

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-140998

(43)Date of publication of application : 02.06.1995

(51)Int.Cl.

G10L 3/00

G10L 3/00

(21)Application number : 05-353142

(71)Applicant : TOSHIBA CORP
TOSHIBA SOFTWARE ENG KK

(22)Date of filing : 28.12.1993

(72)Inventor : HASHIMOTO HIDEKI
NAGATA HITOSHI
SETO SHIGENOBU
TAKEBAYASHI YOICHI
YAMAGUCHI KOJI
SHINCHI HIDEAKI

(30)Priority

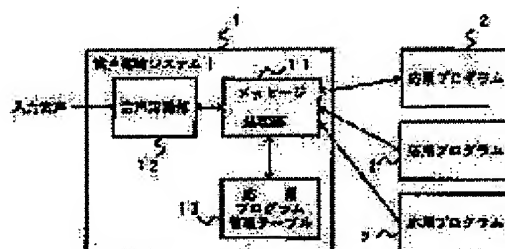
Priority number : 04358597	Priority date : 28.12.1992	Priority country : JP
05 78920	12.03.1993	JP
05256405	20.09.1993	JP

(54) VOICE RECOGNITION INTERFACE

(57)Abstract:

PURPOSE: To provide a voice recognition interface excellent in use convenience, capable of simultaneously handing plural application programs from a voice recognition system.

CONSTITUTION: A voice recognition system 1 is connected to plural application programs 2. The system 1 controls the information relative to the programs 2 by an application program control table 13. Based on the information of the table 13, a message processing section 11 makes decisions on the recognition object vocabularies corresponding to the voice input, sending addresses of the recognition results at a voice identification section 12 which identifies the voice and the voice focus which specifies the object of the voice input to the plural application programs.



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-140998

(43) 公開日 平成7年(1995)6月2日

(51) Int.Cl.⁶

G 1 0 L 3/00

識別記号

5 6 1 H 9379-5H

R

序内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数10 F D (全 58 頁)

(21) 出願番号 特願平5-353142

(22) 出願日 平成5年(1993)12月28日

(31) 優先権主張番号 特願平4-358597

(32) 優先日 平4(1992)12月28日

(33) 優先権主張国 日本 (J P)

(31) 優先権主張番号 特願平5-78920

(32) 優先日 平5(1993)3月12日

(33) 優先権主張国 日本 (J P)

(31) 優先権主張番号 特願平5-256405

(32) 優先日 平5(1993)9月20日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(71) 出願人 000221133

東芝ソフトウェアエンジニアリング株式会
社

東京都青梅市新町1385番地

(72) 発明者 橋本 秀樹

東京都青梅市新町1385番地 東芝ソフトウ
ェアエンジニアリング株式会社内

(74) 代理人 弁理士 鈴江 武彦

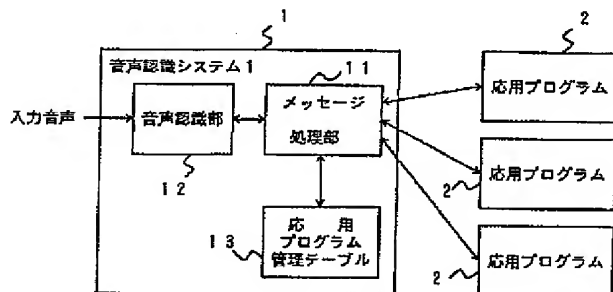
最終頁に続く

(54) 【発明の名称】 音声認識インターフェース

(57) 【要約】

【目的】 本発明は、音声認識システムより複数の応用プログラムを同時に取扱うことが可能で、使い勝手に優れた音声認識インターフェースを提供する。

【構成】 音声認識システム1に複数の応用プログラム2を接続したものであって、音声認識システム1は、応用プログラム管理テーブル13により複数の応用プログラム2に関する情報を管理していて、この応用プログラム管理テーブル13の情報に基づいて音声入力に対する認識対象語彙の決定、音声を識別する音声識別部12での認識結果の送信先の決定、音声入力の対象を複数の応用プログラムに指定する音声フォーカスの決定をメッセージ処理部11により実現している。



【特許請求の範囲】

【請求項1】 音声認識システムに複数の応用プログラムを接続した音声認識インターフェースにおいて、前記音声認識システムは、音声認識する音声認識手段、前記複数の応用プログラムに関する情報を管理する応用プログラム管理手段、この応用プログラム管理手段の情報に基づいて音声入力に対する認識対象語彙の特定、前記音声認識手段の認識結果の送信先の特定を行うメッセージ処理手段を具備したことを特徴とする音声認識インターフェース。

【請求項2】 音声入力の対象を単数または複数の応用プログラムに指定する音声フォーカスの変更を音声入力より可能にする手段を有する請求項1記載の音声認識インターフェース。

【請求項3】 音声入出力システムに複数の応用プログラムを接続した音声入出力インターフェースにおいて、前記音声入出力システムは、音声認識する音声認識手段、音声を合成する音声合成手段、前記複数の応用プログラムに関する情報を管理する応用プログラム管理手段、この応用プログラム管理手段の情報に基づいて音声入力に対する音声認識対語語彙の決定、前記音声認識手段の認識結果の送信先の決定、前記複数の応用プログラムからの音声出力の制御を行なうメッセージ処理手段を具備することを特徴とする音声入出力インターフェース。

【請求項4】 前記音声合成手段は、複数の応用プログラムからの音声データ出力要求に関する情報を管理する音声出力管理手段、文字データから音声データを合成する波形合成手段、前記出力要求に関する情報に従い前記複数の音声データに含まれる同時刻のデータを重畳する音声データ重畳手段を具備することを特徴とする請求項3記載の音声入出力インターフェース。

【請求項5】 音声入出力システムに接続され音声により電子メールの送受信を行なう音声入出力インターフェースにおいて、応用プログラムにおいて行なう作業の重要度を管理するタスク重要度管理手段と、前記作業の重要度と受信した電子メールとの比較を行ない電子メールの受信の通知方法を決定する電子メール通知決定手段とを具備することを特徴とする電子メールシステム。

【請求項6】 請求項1に記載の音声認識システムまたは請求項3に記載の音声入出力システムに複数の応用プログラムと応用プログラムの操作を行なう音声インターフェース管理システムとを接続した音声入出力インターフェースにおいて、前記音声インターフェース管理システムは、前記音声入出力システムからの音声認識結果から前記応用プログラムにおける操作へのメッセージ変換を行うメッセージ変換手段と、前記メッセージ変換に関する情報を管理する音声インターフェース管理手段と、前記音声認識結果に

対応した前記応用プログラムにおける操作を登録するためのプログラム操作登録手段とを具備することを特徴とする音声入出力インターフェース。

【請求項7】 前記応用プログラム管理手段は、応用プログラムを構成するウィンドウごとに音声フォーカスおよび認識対象語彙の管理を行ない、前記メッセージ処理手段は、前記応用プログラム管理手段の情報、ならびに音声フォーカスの情報および認識語彙の属性情報に基づいて、認識結果の送信対象を決定することを特徴とする請求項6に記載の音声入出力インターフェース。

【請求項8】 前記音声インターフェース管理システムは、音声認識に用いる認識辞書の内容を表示してユーザが編集を行なう辞書内容表示編集手段と、辞書内容を検索する辞書内容検索手段とをさらに具備することを特長とする請求項6または7に記載の音声入出力インターフェース。

【請求項9】 前記音声インターフェース管理システムは、応答音声を管理する応答音声管理手段と、応答音声を登録する応答音声登録手段とをさらに具備し、前記メッセージ変換手段は、該応答音声管理手段の応答音声に関する情報を用いて、音声入力に応じた応答音声の出力を行なうことを特徴とする請求項6または8に記載の音声入出力インターフェース。

【請求項10】 前記音声認識システムまたは前記音声入出力システムは、認識辞書作成のための音声データ収集を行なう学習データ処理手段と、認識辞書を作成する辞書作成手段とを具備し、前記学習データ収集手段は収集データの音声区間を音により強調して再生することを特徴とする請求項6または7に記載の音声入出力インターフェース。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、パーソナルコンピュータおよびワークステーションなどに用いられる音声認識インターフェースに関するものである。

【0002】

【従来の技術】近年、計算機はキーボード、マウス、音声、画像など複数の入力手段を装備し、様々な指示やデータ入力を可能にしたものが考えられている。

【0003】このうちで、音声入力は、人間にとって自然であり有力な入力手段といえるが、音声処理にかかる計算量や認識率などの点で問題があり、入力手段として広く利用されるに至らなかった。

【0004】しかして、従来、音声認識インターフェースにおける、応用プログラムと音声認識システムの構成として、次のようなものが考えられている。

【0005】図122は、応用プログラムAPに音声認識システムSR Sが組み込まれているものである。このようにしたものは、音声認識機能が応用プログラムAP

から分離できないため、他の応用プログラムから音声認識機能を利用するのが困難であった。

【0006】また、図123は、一つの音声認識システムSRSと一つの応用プログラムAPからなり、両者が接続される構成のものである。このようにしたものは、音声認識システムSRSは、接続された応用プログラムAPに占有されるため、同じ音声認識システムSRSを別の応用プログラムから利用するには、別の応用プログラムに接続を変更する必要がある、接続し直すための手間がかかる。また、音声認識システムSRSと応用プログラムAPの間でやり取りするデータは、音声認識システムSRSから応用プログラムAPへ送られる認識結果のみであるため、音声認識システムSRSは応用プログラムAPの内部状態を知ることができない。このため、応用プログラムAPの内部状態に応じた認識対象語彙の変更などが自動的に行えず、利用者が語彙の変更を行う必要があるため、使い勝手の悪いシステムになっていた。

【0007】また、図124は、一つの音声認識システムSRSと一つの応用プログラムAPからなり、それらが相互に接続されて、認識語彙や認識結果などの情報を送り合う構成のものである。このようにしたものは、音声認識システムSRSは応用プログラムAPの内部状態や認識語彙などを知ることができるため、認識語彙の変更を自動的に行うことができるが、音声認識システムSRSは応用プログラムAPに占有されるため、同時に他の応用プログラムが音声認識システムSRSを利用することができない。

【0008】また、図125は、文献[Schmandt et al, "Augmenting a window system with speech input", COMPUTER, Vol. 23, pp. 50-58, 1990]のシステムの構成であり、一つの音声認識システムSRSから複数の応用プログラムAPに音声認識結果を一方的に送るものである。このシステムでは、ウインドウシステムを利用し、音声認識結果をマウスやキーボードによる入力に翻訳することによって音声を入力している。この構成のシステムでは、複数の応用プログラムAP音声認識機能を同時に利用できるが、音声認識システムSRSが応用プログラムAPの内部状態を知ることができないため、応用プログラムAPの内部状態に応じた認識処理を行うことができない。

【0009】また、図126は、文献[Rudnicky他, spoken language recognition in an office management domain, Proc. ICASSP '91, S12.12, pp. 829-832, 1991]のシステムの構成であり、一つの音声認識システムSRSと複数の応用プログラムAPからなり、音声認識システムSRSと応用プログラムAPが相互に情報を送り合

って音声認識を行う。このシステムには、複数の応用プログラムが連続音声認識を共用できるという特徴があり、高価な音声認識装置の利用に関して有用な方法を取っているといえるが、リアルタイム処理やワークステーション上での利用形態についての検討は十分ではない。この構成では、複数のプログラムが音声認識機能を利用可能であり、また、応用プログラムAPの内部状態に応じた認識システムSRS側の処理も可能であるが、同時に一つの応用プログラムAPとしか接続できないため、複数の応用プログラムAPを同時に扱えるという音声の特徴を生かした処理を行うことができなかった。また、どの応用プログラムAPに音声認識結果を送るかの決定は、音声認識システムSRSから行われていたため、例えば応用プログラムAP側で認識結果を必要としている場面でも、認識結果を得られない場合があった。

【0010】

【発明が解決しようとする課題】このように従来の音声認識インターフェースによると、応用プログラムAPが音声認識対象を管理できないため、応用プログラムAP主導の音声入力制御ができず、利用者に音声認識を促したい状態でも、音声認識システムSRSからの音声入力許可命令を受けとるまで待たなければならなかった。また、1つの音声で複数の応用プログラムAPを同時に制御することができないため、例えば「終了」という1つの音声入力で、複数の応用プログラムAPを終了させることができなかった。また、認識結果にしたがって、音声入力を複数の応用プログラムAPに振り分けることができないため、音声の入力に先立って入力対象を特定することが必要とされていた。また、1つの音声入力に対して1つの音声認識システムしか動作しないため、例えば孤立単語認識と連続音声認識のように異なる種類の認識方式を共存させ、同時に利用するようなことができなかった。

【0011】本発明は、上記事情に鑑みてなされたもので、音声認識システムより複数の応用プログラムを同時に取扱うことが可能で、使い勝手に優れた音声認識インターフェースを提供することを目的とする。

【0012】

【課題を解決するための手段】本発明は、音声認識システムに複数の応用プログラムを接続した音声認識インターフェースにおいて、音声認識システムは、音声を識別する音声識別手段、前記複数の応用プログラムに関する情報を管理する応用プログラム管理手段、この応用プログラム管理手段の情報に基づいて音声入力に対する認識対象語彙の決定、前記音声識別手段の認識結果の送信先の決定を行うメッセージ処理手段により構成されている。

【0013】

【作用】この結果、本発明によれば各応用プログラムにより音声認識システムに対する音声認識結果の受信の可

否を決定できるので、応用プログラムが自分や他の応用プログラムの音声入力に関する制御を自由に行うことができ、柔軟で使いやすい音声認識インターフェースが構築できる。

【0014】また、音声認識システムがその音声認識結果を同時に複数の応用プログラムに送信できるので、一つの音声入力による操作を同時に複数の応用プログラムに対して行うこともでき、音声入力による計算機の操作性も向上する。

【0015】さらに、音声認識システムが複数の応用プログラムに対する音声認識を行えるので、音声入力対象の明示的な指定をせずに音声認識結果に基づき音声入力を各応用プログラムに振り分けることができ、利用者の負担を軽減できる。

【0016】

【実施例】以下、本発明の実施例を図面に従い説明する。

【0017】（第1実施例）図1は同実施例の概略構成を示している。図において、1は音声認識システムで、この音声認識システム1は、メッセージ処理部11、音声認識部12、応用プログラム管理テーブル13から構成され、メッセージ処理部11に複数の応用プログラム2を接続している。

【0018】この場合、音声認識システム1は、応用プログラム2からのメッセージに含まれる指示に従って音声認識を行い、認識結果をメッセージとして応用プログラム2に送る。応用プログラム2は、その音声認識結果を利用してその応用に依存した固有の処理を行う。また、音声認識システム1は、同時に複数の応用プログラム2とメッセージを交換し音声認識結果を送信できるようにしている。

【0019】音声認識システム1を構成するメッセージ処理部11は、応用プログラム2と音声認識部12のメッセージを交換し、音声認識システム1の全体制御を行う。また、音声認識部12は、メッセージ処理部11とメッセージを交換し合うことでメッセージ処理部11から送られてくる情報に従って入力音声に対して音声認識を行い、その結果情報をメッセージ処理部11に通知する。

【0020】応用プログラム管理テーブル13は、音声認識システム1と通信を行う全ての応用プログラム2に関する情報を収納するテーブルである。このテーブルは、音声が入力された際の認識対象語彙の決定や、認識結果の送信先の決定に利用され、これにより音声認識システム1は同時に複数の応用プログラム2とのメッセージ交換を行うことができる。また、応用プログラム管理テーブル13は、プログラムID、入力マスク、認識対象語彙リスト、音声入力フラグを持っている。プログラムIDは、音声認識システム1により応用プログラム2に対して一意に付けられる識別のための番号である。入

力マスクは、音声認識システム1から応用プログラム2に送信するメッセージの種類を限定するものである。認識語彙リストは、応用プログラム2が音声認識システム1に対して要求した認識語彙が記述されるテーブルである。音声入力時の認識対象語彙の決定に利用される。音声入力フラグは、当該応用プログラム2に音声フォーカスが当たっているか否かを表している。なお、応用プログラム2に音声フォーカスが当たるという言葉は、応用プログラム2が音声入力対象となるということを意味するものとする。すなわち、音声フォーカスは、認識結果の送信対象を特定するものである。

【0021】図2は音声認識部12の概略構成を示している。

【0022】この場合、音声認識部12は、音声検出部121、音声分析部122、認識辞書照合部123および音声認識辞書124からなっている。

【0023】音声検出部121は、例えば一定時間間隔ごとの入力音声のパワーをもとにして検出を行う方法（永田、他“ワークステーションにおける音声認識機能の開発”，電子情報通信学会技術報告、HC9119，pp. 63-70，（1991））が知られている。音声分析部122は、音声検出部121で検出される音声区間に対して、例えばFFTやバンドパスフィルタなどを用いて周波数分析を行い、単語音声の特徴パラメータを抽出する。認識辞書照合部123は、音声分析部122からの出力パラメータを用いて、例えば複合類似度法（上記の研究資料）やHMM、DPマッチングなどの手法により認識辞書124との照合を行い、スコアの最も高い語彙を認識結果として出力する。

【0024】そして、認識辞書照合部123では、音声特徴パラメータと認識辞書124と照合を行う際、照合前に無駄な処理を行わないため、その時点で認識辞書124のどの語彙と照合を行うべきかをメッセージ処理部11に問い合わせ、その照合情報にしたがって認識辞書124との照合処理を行う。そして、認識の成功失敗にかかわらずその認識結果はメッセージ処理部11に送られ、応用プログラム管理テーブル13の内容にしたがって応用プログラム2に認識結果を送るようになる。

【0025】ここで、図2では、認識部の要素が全て一体となっており、1つのプロセスとして動作可能であるが、図3に示すように音声検出部121を分離した構成も可能である。音声検出部121と後続の音声分析部122、認識辞書照合部123を、例えばべつプロセスとして両者の間のデータのやり取りをプロセス間通信により行えば、音声検出部121を独立した形で扱うことができ、例えば、図4に示すように複数の音声検出部121からの出力を共通の音声分析部122、認識辞書照合部123で扱うことができる。また、図5に示すように音声検出部121と音声分析部122を一体にして、認識辞書照合部123と認識辞書124を分離した構成も

可能である。

【0026】図6は応用プログラム2の概略構成を示している。

【0027】この場合、応用プログラム2は、メッセージ入出力部21およびプログラム本体22からなっている。メッセージ入出力部21は、音声認識システム1とメッセージ交換を一括して行うもので、音声入力の手続きを応用プログラム2の作成者に提供するものである。また、複雑なメッセージ送受信規約を応用プログラム作成者から隠蔽し、全ての応用プログラム作成者に統一的に通信手続を提供するためでもある。プログラム本体22は、応用プログラムに依存した処理の手続を行うプログラムであり、応用プログラム固有の内部状態に従った音声認識システム1に対する命令や、音声認識システム1から音声認識結果を受け取った際の手続などを含んでいる。

【0028】次に、このように構成した実施例の動作を説明する。

【0029】この場合、音声認識システム1と応用プログラム2との間の情報のやり取りは、メッセージ交換によって行う。ここで、メッセージとは、ある構成要素からほかの構成要素に渡されるコマンドやそのコマンドの実行結果、音声認識結果などのデータを総称している。

【0030】メッセージによる通信は、例えば音声認識システム1をサーバ化し、また応用プログラム2を音声認識システムのクライアントとし、その間をT P C、D E C n e t、S t r e a mなどのバイトストリーム型のプロトコルを利用して実装する。音声認識インターフェースの各構成要素間で交換されるメッセージを、次に説明する図7に示している。これらのメッセージの処理は、すべて音声認識システムのメッセージ処理部11が担当する。なお、上述の実施例では、図1の音声認識システム、全体が1つのプロセスとして実行するものとして説明したが、音声認識システムの構成要素である音声認識部、メッセージ処理部、応用プログラム管理テーブル、それぞれを別個のプログラムとして実行することも可能である。

【0031】[音声認識システム1と応用プログラム2の間のメッセージ] 応用プログラム2から音声認識システム1へのメッセージは、図7(a)に示すような種類がある。これらは、基本的に、応用プログラム2から音声認識システム1への命令を意味している。

【0032】ここで、通信路接続/切断要求は、応用プログラム2が音声認識システム1とメッセージを交換するにあたって、その通信路を接続/解放する要求である。音声認識辞書のロード/解放要求は、応用プログラム2が利用したい語彙を含む音声認識辞書を音声認識システム1にロード/解放する要求である。認識語彙設定要求は、応用プログラム2が、どの認識辞書のどの語彙を使って認識を行うかを音声認識システム1に要求する

ものである。入力マスク設定要求は、応用プログラム2が、音声認識システム1から受け取りたいメッセージの種類を設定する要求である。入力タスク設定要求は、音声フォーカスを、指定した応用プログラム2に変更する要求である。認識開始/終了要求は、音声認識システム1に対する音声認識開始/終了の要求である。

【0033】一方、音声認識システム1から応用プログラム2へのメッセージは、図7(b)に示すような種類があり、2つに分類できる。1つは、応用プログラム2からの命令やデータの問い合わせなどの要求に対する応答で、これは上記の要求メッセージに対応する。もう1つのメッセージは、音声認識結果の情報や、音声認識システムの内部状態の変化にともない、音声認識システムによって生成されるメッセージである。

【0034】ここで、音声認識結果は、音声認識システム1が、応用プログラム2の設定要求をした認識語彙を用いて認識した結果を通知するメッセージである。認識が成功した場合には、すくなくとも1つの認識語彙を含み、その語彙が何であるか、その語彙の持つ辞書はどれか、認識処理結果としての得点などの情報を含む。失敗した場合には(音声レベルが高すぎたとか低すぎたなど)、失敗した原因に関する情報を持っている。入力タスク変更通知は、入力タスク設定要求などで音声フォーカスが実際に変更された際に応用プログラム2に対して送信されるメッセージで、変更される前のタスクIDおよび変更後のタスクIDを含んでいる。認識辞書ロード/解放通知は、認識辞書ロード/解放要求などで認識辞書が新たにロードされたり解放された場合に送信されるメッセージである。通信路接続/切断の通知は、応用プログラム2が音声認識システム1に対して通信路接続/切断要求を発行した際に発生するメッセージである。応用プログラム2が要求せずに一方的に通信路を切断した場合にも発生する。認識語彙変更通知は、認識語彙設定要求により各応用プログラムの認識語彙が変更された場合に発生するメッセージである。

【0035】これらは、音声入力を受け付け音声認識を行ったときや、音声フォーカスが変更されたときや、応用プログラム2が音声認識システム1と接続したときや、認識語彙が変更されたときなど、音声認識システム1から全ての応用プログラム2に送信できるものであるが、全てのメッセージを応用プログラム2が常時受け取る必要はない。応用プログラム2が、どのメッセージを受け取るかの設定は、各メッセージに対応する入力マスクを音声認識システム1に通知する事で行う(入力マスク設定要求)。これによって応用プログラム2は、自分の必要とするメッセージのみを音声認識システム1に通知してもらうことができる。

【0036】図8は、入力マスクの種類を示している。これらは応用プログラム2が受け取りたいメッセージの種類に対応しており、同時に複数のマスクが設定できる

ものとする。

【0037】この設定を音声認識システム1に通知することで、入力マスクに対応するメッセージが音声認識システム1の内部で生成される度にそれを受け取ることができる。例えば、音声認識結果マスクを設定すれば、音声入力となされる度に音声認識結果を得ることができるようになるし、入力タスク変更マスクを設定すれば、音声フォーカスが変更される度に、それが応用プログラムに通知されるようになる。

【0038】音声認識システム1と応用プログラム2の間のメッセージとして、上記の2種類のメッセージ(要求メッセージと応答メッセージ)以外に、エラーメッセージが考えられる。エラーメッセージは、成功時には応答を要しない応用プログラム2からの片道メッセージの失敗や、認識システムにクリティカルな状態が発生したときにそれを通知するメッセージである。また、上述したメッセージ以外にも、音声認識システム1の内部情報にアクセスするためのメッセージや、音声入力レベルを変更するなど、音声認識システム1や音声入出力の設定をするメッセージなどさまざまなメッセージが考えられる。

【0039】このように、応用プログラム2は、音声認識システム1の内部状態の変化をメッセージの形で通知させることができるため、それに基づいて音声認識システム1を制御し、さらには他の応用プログラム2が制御できるようになるため、自由度の高い、柔軟なインターフェースを音声によって制御することができる。

【0040】さて、音声認識システム1は、メッセージ処理部11と音声認識部12を有しているが、これらの間もメッセージによって情報交換がなされる。なお、音声認識システム1における応用プログラム2とのメッセージは、その全てをメッセージ処理部11が取扱っている。

【0041】[音声認識部12とメッセージ処理部11の間のメッセージ] 音声認識部12からメッセージ処理部11へのメッセージは、図7(c)に示す種類がある。ここで、認識語彙照会要求は、音声認識システムに音声が入力された時点で、入力音声とどの認識語彙との照合を行うべきかを決定するために発行される要求である。音声認識結果は、入力音声とその時点で認識すべき認識語彙との照合結果をメッセージ処理部11に通知するものである。

【0042】一方、メッセージ処理部11から音声認識部12へのメッセージは、図7(d)に示すような種類がある。ここで、認識辞書ロード/解放要求は、応用プログラム2が音声認識システム1に対して発行した認識辞書ロード/解放要求がそのまま音声認識部12へ引き渡された所のメッセージである。認識語彙情報は、音声認識部12からメッセージ処理部11への認識語彙照会要求に対する応答である。

【0043】このようにして、音声認識システムを構成する各部において、メッセージをやりとりすることで、処理が進められるが、次に、音声認識インターフェースとして、処理がどのように進められていくかを図9に従い説明する。同図では、応用プログラム2が起動されてから、最初に音声認識結果を受け取るまでのタイムチャートを示している。

【0044】この場合、応用プログラム2は、まず音声認識システム1との接続要求(a)を送る。接続が達成されたならば、音声認識語彙を含む認識辞書ロード要求(b)と、ロードした辞書中で音声入力に使いたい語彙を認識語彙とする設定要求(c)を発行する。メッセージ処理部11では、(a)に対しては応用プログラム2との通信路接続処理を行って、その結果を応用プログラム2に返す。(b)に対しては、メッセージをそのまま音声認識部12に送って辞書がロードされるのを待ち、辞書のロードの結果を応用プログラム2に返す。(c)に対しては指定された認識語彙を応用プログラム管理テーブル13に書込み、その処理結果を返す。認識対象語彙が無事に設定されたならば、応用プログラム2は、入力マスク設定要求(d)と入力タスク設定要求(e)を送る。メッセージ処理部11では、(d)と(e)を受けて、それぞれ応用プログラム管理テーブル13に書込む。

【0045】以上が、音声認識システム1に対する応用プログラム2からの初期設定要求となる。初期設定が終わったならば、音声認識システム1からのメッセージ待ちに入る。メッセージを待ちながら、応用プログラム2固有のタスクに依存した処理などを行う。処理に伴う内部状態の遷移などに従い、認識語彙を変更する要求や入力タスクを自分自身や他の応用プログラム2に変更する要求など、任意の要求を自分の処理に応じて音声認識システム1に送り、音声認識システム1を応用プログラム2側から制御できるようにしている。

【0046】ここで、音声入力に応用プログラム2に対して行われたとする。すると入力音声は、まず音声認識部12において、音声区間の検出と分析が行なわれる。音声認識部12は、音声分析を済ませたならば、その時点で認識対象となっている語彙を知るために、メッセージ処理部11に対して認識語彙照会要求(f)を送る。メッセージ処理部11では、これを受信すると、応用プログラム管理テーブル13を参照してこの場面で音声認識処理を行うべき語彙を調べ、その結果である認識語彙情報を音声認識部12に返す。音声認識部12では、

(g)により指定された認識対象語彙に対応する認識辞書データと分析済みの分析データを照合し、その結果をメッセージ処理部11に送る。メッセージ処理部11では、(g)のうちの1位の尤度をもつ語彙を応用プログラム管理テーブル13の認識対象語彙中で探し、それを持つ応用プログラム2の音声入力フラグが1であり、か

つ入力マスクとして認識結果通知マスクが設定されているならば、その応用プログラムに対して認識結果を送信する。

【0047】図9で説明した処理を、さらに具体例を用いて説明する。

【0048】音声認識システム1と接続している応用プログラム2がシェルツールとテキストエディタの2つであった場合の応用プログラム管理テーブル13は、図10(a)に示すようになる。

【0049】ここで新しくメールツールを起動する際の処理を説明する。起動されたメールツールが、まず通信路接続要求(a)を送信すると、応用プログラム管理テーブル13にメールツール用の領域が取られ、メールツールのプログラムIDが付けられる。プログラムIDは、例えば応用プログラム2の起動順に0から付けられるとする。次に認識辞書ロード要求(b)を送る。ここでは認識辞書はすでにロードされており、音声認識システム1は、そのことを応用プログラム2に知らせる。次に、認識語彙設定要求(c)で認識語彙として「先頭」「最後」「前」「次」「送信」「終了」を送り、入力マスクとして認識結果通知マスクを送る(d)。入力タスク設定要求(e)として、現在当たっている全ての音声フォーカスを無効にし、音声フォーカスをメールツールに当てる要求をする。

【0050】なお、本実施例では、1つの認識辞書をすべての応用プログラム2で共通に使うこととし、従って、図10においては、複数の辞書を利用する場合に必要な各語彙がどの辞書に含まれるかを示す情報を省略する。

【0051】以上の処理により、応用プログラム管理テーブル13は、図10(b)のようになり、シェルツールに当たっていた音声フォーカスは、新たに起動されたメールツールに変更され、メールツールは音声入力が可能となる状態になる。

【0052】ここで、例えば「次」という音声が入力されたとする。入力された音声は、音声認識部12において音声区間検出と分析処理を施され、音声特徴パラメータが求められる。音声認識部12は、この音声特徴パラメータと照合する辞書データを知るべく、メッセージ処理部11に対して認識語彙照合要求(f)を送る。この要求を受けたメッセージ処理部11は、応用プログラム管理テーブル13を参照してその時点での認識対象語彙を知る。ここでは、音声入力フラグが1であり、かつ入力マスクに認識結果通知マスクが設定されているメールツールの認識対象語彙リスト中の全ての語彙「先頭」「最後」「前」「次」「送信」「終了」がその時点で入力可能な語彙となる。これら6つの語彙が音声認識部12に通知され、音声認識部12は、これら語彙に関する辞書データと分析された特徴パラメータに対して照合処理を行い、その結果をメッセージ処理部11に送る

(g)。メッセージ処理部11は、認識結果を受けとると、応用プログラム2中の音声入力フラグが1であり、かつ入力マスクに認識結果通知マスクが設定されている応用プログラム2の認識対象語彙リスト中に認識結果の語彙を探し、発見したならばその認識結果をその語彙リストを持つ応用プログラム2に対して送信する。

【0053】先の音声入力の認識結果が「次」であった場合には、メールツールに送信されることになる。

「次」という認識結果をメッセージ入出力部21を介して受けとった応用プログラム2は、例えば現在表示している受信メールの次のメールを表示するといった処理を行う。

【0054】図10(a)(b)では、シェルツールの入力マスクとして、認識結果通知マスクが設定されている。このマスクにより音声フォーカスの変更が発生する度にそれが通知されるようになる。

【0055】上述の例では、メールツールからの入力タスク設定要求(e)を音声認識システム1が受信し、メッセージ処理部11が音声フォーカスの変更を行ったときに入力タスク変更通知のメッセージがシェルツールに送られる。認識結果通知マスク以外の入力マスクは音声入力フラグの値に依存していないため、入力タスク変更マスクが設定されていれば、音声入力フラグの値に関係なく、音声フォーカスの変更メッセージが、それが起きる度に応用プログラム2に通知される。応用プログラム2は、このような音声認識システム1の内部状態の変化をメッセージを介して知ること、様々な柔軟な処理をすることができる。例えば、シェルツールは、音声フォーカスを失ったことを利用者に画面表示や合成音声またビープ音などを通じて知らせることができる。

【0056】このようにして、応用プログラム2は、メッセージを通じて音声認識システム1を自由に制御できるようになり、応用プログラム主導の柔軟な音声認識インターフェースが得られることになる。

【0057】従って、第1実施例によれば、複数の応用プログラム2が同時に平行して動作するマルチタスク環境において、各応用プログラム2が音声認識システム1と通信により直接メッセージ交換を行い、認識語彙や認識結果などのデータを直接相互に交換できるため、全ての応用プログラム2にキーボードやマウスなどの様に、音声入力を標準的な入力手段として装備することができるため、ワークステーションなどのマルチタスク環境における音声入力の本格的な利用が可能となり、音声を含めたマンマシンインターフェースの使い勝手の向上が期待できることになる。

【0058】なお、本実施例は、孤立単語認識を応用した音声認識インターフェースの実施例であったが、連続単語音声認識や連続音声認識を応用することも可能である。

(第2実施例) 同第2実施例では、マルチタスクの計算

機環境において、ウィンドウシステムを同時に利用することで、ユーザの利用環境の向上を可能にしている。

【0059】ウィンドウシステムを同時に利用する場合の構成を図11に示している。この場合、音声入力を扱う音声認識システム3と、キーボード入力およびマウス入力を扱うウィンドウシステム4と、これら音声認識システム3およびウィンドウシステム4と相互にメッセージを通信する1つ以上の応用プログラム5からなっている。つまり、同実施例では、上述の第1実施例にウィンドウシステムを追加し、応用プログラムにウィンドウシステムとの間の通信手段を持たせるようにしている。

【0060】ウィンドウシステム4と音声認識システム3は、相互に独立している。また、ウィンドウシステム4と応用プログラム5との間のメッセージは、マルチウィンドウ環境におけるウィンドウの生成や、キーボード入力やマウス入力などの処理に関するものである。

【0061】本実施例を説明する前に、マルチウィンドウを実現するウィンドウシステムについて簡単に説明する。ワークステーションなどのマルチタスクの計算機環境でマルチウィンドウを実現するウィンドウシステムは、その環境下で動作する複数の応用プログラムと通信し、各々の応用プログラムをビットマップディスプレイと呼ばれる表示画面に抽象化して表示される。そこでは、応用プログラムごとに基本的に1つのウィンドウが割り当てられる。

【0062】図12は、一般的なウィンドウシステムの画面表示例である。この例では、A、B、Cの3つの応用プログラムが平行して動作している。ウィンドウシステムは、キーボードやマウスなどの入力装置を管理し、複数の応用プログラムに入力装置を共有させる。マウス画面中では、矢印型のマウスポインタとして抽象化されており、ウィンドウの操作や入力対象の指定などに使われる。

【0063】なお、本願の実施例では専ら、ポインティングデバイスとしてマウスを利用した説明をしているが、ペンやタッチパネルなど他のポインティングデバイスを用いることも可能であり、全ての実施例における記述はこれら他のポインティングデバイスについても全く同様に適用できる。

【0064】キーボード入力を行う対象は、キーボードフォーカスによる。キーボードフォーカスは一般的にマウスポインタによって指定される。キーボードフォーカスの当たっている応用プログラムは、ウィンドウ枠がそれ以外のウィンドウより太くしたり、ウィンドウ上部のタイトルバーの色を変えることで表現される。図12では、応用プログラムBにキーボードフォーカスが当たっている様子を示す。キーボードフォーカスは一般に常に1つのウィンドウにだけ当てられる。

【0065】ここで、第1実施例で述べた3つのプログラム、つまりシェルツール、テキストディタ、メールツ

ールを再び利用して説明する。この場合、各プログラムは、ウィンドウシステムによって各々1つのウィンドウとして抽象化され表現される。また、音声認識システムとも通信を行い、起動時に音声認識システムに対して第1実施例で示した手順をもって認識語彙を設定する。各応用プログラムの認識語彙は同じく図13に示す通りである。

【0066】一般に既存のウィンドウシステムにおいて、応用プログラムは、キーボードフォーカスの変更の通知が受け取られる。キーボードの入力対象と音声の入力対象を同じ応用プログラムにするために、応用プログラムはキーボードフォーカスが当たったならば、音声認識システムに対し、自身に音声フォーカスを当てる要求をし、外れたならば音声フォーカスをはずす要求をする。これは、第1実施例で述べた、入力タスク変更要求を送信することで可能となる。以下ではキーボードフォーカスと音声フォーカスを一致したものとして扱い、それを入力フォーカスと呼ぶ。入力フォーカスはマウスによって操作する。

【0067】入力フォーカスの移動に伴う音声認識語彙の変化を図14に示している。この場合、図14(a)は状態1、図14(b)は状態2を示すもので、入力フォーカス(それと同時に音声フォーカスも)がテキストエディタに当たっている。従って、この状態で認識可能な語彙は、テキストエディタの認識語彙である、「カット」「コピー」「ペースト」「解消」「終了」の5つである。ここではユーザはこの5つの語彙を発声すると、音声認識結果がテキストエディタに送られるということである。マウスポインタによりシェルツールを指定すると、入力フォーカスはシェルツールに移動し(それと同時に音声フォーカスもシェルツールに移動し)認識可能な語彙は、シェルツールの認識語彙である「ヒストリ」「リスト」「ホーム」「プロセス」「終了」の5つに変化する。

【0068】音声認識語彙として何を使うかは自由であり、応用プログラム毎の認識語彙をユーザが記憶、判断することは、ユーザへの大きな負担となる。しかし、個々の応用プログラムに認識語彙の表示を可能にする手段をもたせるのは逆に応用プログラムの作成者にとって負担となる。また、音声入力は、キーボードなどの入力手段と違って曖昧性があるため、入力音声が正しく認識されたかをユーザが確認できることが重要となってくる。

【0069】この問題を解決する手段として、音声認識インタフェースに標準的な応用プログラムとして、図15に示すような、認識語彙を表示するプログラム(語彙表示プログラム)を作成することが考えられる。このプログラムは、全ての応用プログラムが、新規の応用プログラムが通信路を接続/切断したり、語彙の変更を要求したり、音声フォーカスの変更をする度に、自身に、それらの要求により発生するメッセージを送信するように

要求する(すなわちそれを受け取る入力マスクを設定する)。語彙表示プログラムは、常に、その時点で認識可能な語彙をすべて表示できる。また、音声認識される度に、それを知り、応用プログラムに送信された認識結果を、例えば図15のように色を変えて表示することで、音声認識システムが受け付けた音声入力を確認できる。認識語彙表示プログラムにより、応用プログラムの使用者と作成者の双方の負担を軽減し、より使いやすい音声入力環境をユーザに提供できる。

【0070】また、語彙表示プログラムのリスト中の色を変える以外に、認識結果は別の方法によっても、使用者に通知できる。

【0071】例えば、ディスプレイのスクリーン、あるいはアプリケーションのウインドウの特定の位置に認識結果を表示させる方法もある。この表示部分は、アプリケーションごとに持つことも音声認識システム自身が所有していてもよい。ウインドウシステム環境下では、認識結果表示用のウインドウを作成しておき、アプリケーションのウインドウの中央部や上下左右などの周囲の部分あるいはマウスなどのポインタ、キーボード入力のカーソルなどの付近など、特定の位置に表示させるように位置調節をすればよい。

【0072】また認識結果は、次の認識結果が得られるまで表示し続けてもよいし、認識結果が得られた直後だけ表示し、ある時間が経過した後は、次の認識結果が得られるまで表示させないようにしておいてもよい。特にマウスなどのポインタやキーボード入力のカーソルの付近は、視線の移動がわずかで済む利点がある反面、作業している領域の近くで常時表示させると作業の妨げになる場合もあるので、認識結果が得られた直後だけ表示するのは有効である。これとスクリーンやアプリケーションの特定位置に認識結果を常時表示させる方法と併用してもよい。

【0073】応用プログラム間だけではなく、1つの応用プログラム内でも、そのマウスの位置によって音声認識語彙を変更することで、必要以上の認識処理を減らし、音声入力をより確実なものにできる。例えば、図16(a)および同図(b)に示すように、メーリングツールをリスト表示部とテキスト表示部の2つに分割し、そのどちらにマウスポインタがあるかによって認識語彙(ここでは認識語彙は8つである)を変える。こうすることで、必要以上の無駄な認識処理をおさえるとともに、入力音声の認識誤りを起こりにくくすることができるという効果がある。

【0074】また、第1実施例では、新たなアプリケーションが起動されると、そのアプリケーションに音声フォーカスに移ることを説明した。同様にアプリケーションの起動、終了時あるいは、マウス、ペンなどのポインティングデバイス、キーボードなどの入力操作や音声認識の結果を受けて実行される処理の結果として、アプリ

ケーションのウインドウ状態変化がある(ウインドウが生成破壊、ジオメトリ変更された)場合には、音声フォーカスの移動を行う規則を作ることにより使い勝手を向上させることができる。

【0075】例えば、「ウインドウの破壊、アイコン化、ウインドウが他のウインドウに隠れる、等の場合音声フォーカスを失い、ウインドウの生成、非表示状態から表示状態への変更、ウインドウが他のウインドウの上に表示される、ウインドウの大きさを大きくする、等の場合音声フォーカスを獲得する。」、というような規則に従い、各アプリケーション内部でウインドウ状態変化に応じて、フォーカスの獲得・消去を行う。勿論、このような、ウインドウ状態変化は個々のアプリケーションが個別に管理しなくても、音声フォーカスの管理を行うプログラムにより一括管理してもよい。この場合、この管理プログラムは、ウインドウシステムを管理するプログラム(例えば、システムのウインドウサーバ)に管理したいアプリケーションのウインドウの状態変化を知らせてもらい、その通知を受けた時に上述のような規則を適用して、音声フォーカスを変更すればよい。

【0076】また、音声フォーカス管理プログラムがあれば、音声フォーカスを獲得していたアプリケーションがアプリケーションの終了、ウインドウの破壊などにもない音声フォーカスを失った場合も、どのアプリケーションに音声フォーカスを移すかに関して、同様に規則を作り、使い勝手を向上させることができる。

【0077】例えば、「音声フォーカスの履歴を音声フォーカス管理プログラムが保持しておき、音声フォーカスを獲得していたアプリケーションが音声フォーカスを消失した場合、その消失原因が他のアプリケーションのフォーカス獲得要求によるものでないならば、それ以前に音声フォーカスを獲得していたアプリケーションにフォーカスを戻す。」、という規則を作り、音声フォーカス管理プログラムがこれに従って、音声フォーカスを変更させれば、音声フォーカスを獲得しているアプリケーションが1つもない状態、すなわち音声認識システムの出力をどのアプリケーションも受け取らない状態を回避することができる。

【0078】なお、本実施例においては、音声認識システムとウインドウシステムを独立した構成としたが、両システムを統合した形態の音声認識インターフェースの実現も可能である。

【0079】(第3実施例) 第2実施例では、音声認識システムとウインドウシステムを組み合わせ、音声フォーカスとキーボードフォーカスを一致させて、1つの入力フォーカスとし、入力フォーカスをマウスポインタで指定することで、音声認識対象語彙を変更した。しかし、これでは入力フォーカスを変更する度にキーボードから手を離さなければならない。入力フォーカスの変更を音声で可能にすることでユーザはキーボードから手を

離さずに入力タスクを変更し、ユーザのマルチウインドウ環境における使い勝手の向上を期待できる。

【0080】入力フォーカスを音声入力に変更可能にするため、実施例1を拡張して各認識語彙に対してローカルとグローバルの2つの値を設定できるようにする。ローカルな認識語彙とは、それによる認識設定をした応用プログラムに音声フォーカスが当たっている際に認識するような語彙であり、グローバルな認識語彙とは、音声フォーカスがどの応用プログラムに当たっているにかかわらず、認識対象となるような語彙である。

【0081】ここで、再び3つの応用プログラム（シェルツール、テキストエディタ、メールツール）を使って説明する。

【0082】各応用プログラムの認識語彙は、図17に示す通りである。ローカル／グローバル設定にともない応用プログラム管理テーブル中の認識対象語彙リスト中の語彙のそれぞれにローカル／グローバルを示すフラグを設ける。応用プログラム管理テーブルは、図18に示すようになる。音声入力を与えられたときに、メッセージ処理部は、この応用プログラム管理テーブルを使って認識語彙を次のように求める。まず、応用管理テーブルを参照して音声フォーカスの当たっている応用プログラムのローカル認識語彙を拾い出す。次いで全ての応用プログラムのグローバル認識語彙を拾い集める。これらがその時点での認識システムが認識可能とする語彙である。例えば、テキストエディタに音声フォーカスが当たっているとすると、その時点での認識語彙は「カット」「コピー」「ペースト」「取消し」「終了」「シェルツール」「メールツール」「テキストエディタ」の8つである。ここで、「カット」「コピー」「ペースト」「取消し」「終了」「テキストエディタ」の発声に対する認識結果は、テキストエディタに送付され、「メールツール」「シェルツール」は、それぞれメールツール、シェルツールに送信される。例えば、この状態でメールツールを発声した時に、メールツールの中で入力フォーカス（音声フォーカスとキーボードフォーカス）を自分自身に変更すれば、音声入力とキーボード入力の対象をキーボードから手を離すことなく変更できる。

【0083】これは、換言すれば、ウインドウに名前を付けると言うことである。このウインドウ名はウインドウの上部のタイトル表示部に表示すれば、ユーザは、それによってウインドウを何と呼べばよいかわかる。

【0084】以上のように、本実施例では、認識語彙にローカル／グローバルの属性を与えることで、ウインドウに名前を付け、その名前を発声することで手を使わずにフォーカスの変更が可能となり、応用プログラムを切り替えることが可能となる。

（第4実施例）第2、3実施例では、音声フォーカスとキーボードフォーカスを一致させ、同時に1つのウインドウだけが両者の入力を排他的に受け付けるようにし

た。

【0085】この2つの入力フォーカスを一致させることで1つの応用プログラムが両方の入力から一手に引き受けることができた反面、2つの入力手段がありながらそれぞれ別々の応用プログラムに対する入力をできなかった。本実施例では、この2つのフォーカスを分離するために音声フォーカスをマウスポインタによって直接操作しないようにする（キーボードフォーカスは、マウスポインタを使う。）。マウスポインタがウインドウに入り、それが応用プログラムに通知されても、応用プログラムは音声フォーカスを移動させない。この場合は、音声フォーカスは、第3実施例で述べたようにウインドウに名前を付け、それぞれグローバル認識語彙とし、その名前で発声することで変更できる。

【0086】入力フォーカスを分離した際、その2つのフォーカスをユーザに分かりやすく呈示しなければ、ユーザが入力する際に混乱してしまう。本実施例では、キーボードフォーカスをウインドウ枠を太くすることで表示し、音声フォーカスをウインドウタイトルの色を変化させることで示すこととする。

【0087】図19は、入力フォーカスを2つに分離し、それぞれを別々に移動させた場合の例である。同図（a）では、両フォーカスは、どちらもテキストエディタに当たっている。メールツールをマウスポインタで指定すると、キーボードフォーカスはメールツールに移動するが、音声フォーカスは、テキストエディタに当たったままである同図（b）。同図（a）の状態から、「メールツール」音声入力を行うと、音声フォーカスがメールツールに移動するが、キーボードフォーカスはそのままである。同図（b）および（c）において、キーボードフォーカスと音声フォーカスは、それぞれ個別の応用プログラムに当たっているため、まったく同時に別々の入力チャンネルを通じて2つの応用プログラムを操作できる。例えば、同図（c）の状態にすることで、テキストエディタに対してキーボードで文章を打ち込みながらメールツールを音声で操作し、受信した電子メールを読むことができる。

【0088】また、音声フォーカスをコントロールする応用プログラム、音声フォーカスマネージャを作成し、これにより音声フォーカスを音声以外の手段で移動できるようにもしている。図19の右側が音声フォーカスマネージャを示しており、この音声フォーカスマネージャは、同時に動作している応用プログラムの状態を音声認識システムと通信することで知り、リストなどの形で表示する。

【0089】音声フォーカスは、例えば応用プログラム名を反転表示することで表現し、これらリスト上をマウスポインタで指定することで音声フォーカスを変更できるようにする。また、応用プログラムに入力可能な手段は、キーボードや音声以外にもペンなども考えられる。

応用プログラムに入力可能な手段および何が入力できるかを表示すればユーザの使い勝手が向上できる。例えば、入力可能性を手段別にアイコン化することで表示する。

【0090】このように、音声入力対象と音声以外の手段による入力対象を別々に分離することで、複数の入力手段を複数の応用プログラムに割り当て、人間が自然の形の作業を平行して行うことができるようになる。

【0091】(第5実施例) 図20は、同実施例の概略構成を示している。この場合、音声認識システム6に対して複数の応用プログラム7を接続している。そして、これら応用プログラム7には、それぞれメッセージ入出力部71を有している。

【0092】しかして、音声認識システム6は、音声入力があるごとに、その音声に対して認識処理を行い、その認識結果を応用プログラム7に送信する。応用プログラム7は、音声認識システム6に対して認識対象語彙を通知し、音声認識システム6は、それを用いて認識処理した結果を応用プログラム7に送信する。

【0093】応用プログラム7は、メッセージ入出力部71を有していて、このメッセージ入出力部71は、応用プログラム7が認識結果を受け取るか否かを決定し、その要求を音声認識システム6に対して行う。メッセージ入出力部71は、応用プログラム7の指示によって音声認識システム6に対して応用プログラム7のための音声認識を行う要求をしたり、音声認識システム6から送信された認識結果を受けて応用プログラム7に渡したり、ブロックして渡さなかったりする。また、認識対象語彙を変更できる。

【0094】応用プログラム7がメッセージ入出力部71を持つことで、応用プログラム7は、外部からの働き掛けによらず、自分の状態にしたがって音声入力(認識結果)を受けとったり、受けとらなかったりできる。

【0095】例えば、音声による制御が可能な電子メールシステム(音声メールと称する。)の例を挙げると、音声の誤認識による誤動作を防止するため、音声入力が不能な状態で音声メールを起動、動作させておく。音声メールがメールをうけると、例えば「新しいメールを受信しました。いまずぐお読みになりますか。」と合成音声を出力して知らせ、「はい」「いいえ」などの確認を取るための認識対象語彙と、それにより音声認識することを音声認識システム6に通知する。ユーザが「はい」といったならば、新しく受信したメールを表示したり、合成音声によりメールを読み上げたりする。「いいえ」といったならば、音声メールは音声認識システム6に対して音声認識結果を受けとらないように要求し、元の状態に戻る。

【0096】「新しくメッセージを…」のメッセージは、合成音声でなく、図21のように表示してもよい。同図での「はい」「いいえ」は、マウスなどでも操作を

可能とするためである。

【0097】また、図20において、1つの応用プログラム7のメッセージ入出力部71に、他の応用プログラム7の音声入力を可能にしたり、ブロックしたりする機能を与えれば、電子メールの例で言えば、確認のための音声入力を待つ間、電子メールは、他の音声によって制御が可能な応用プログラム7の音声入力を一時的にブロックし、確認が終わった時に戻すような操作が可能になる。

【0098】応用プログラム7による、こうした他応用プログラム7の音声入力をブロックする操作が競合した場合には、時間的に後にブロックモードになった応用プログラム7は、先にブロックモードになった応用プログラム7のブロック解除を待つ操作ができる。

【0099】このように音声認識システム6でなく、応用プログラム7にタスクの管理を可能にする手段を持たせることにより、応用プログラム7が音声認識システム6の指示に従うだけでなく、応用プログラム7独自の内容状態に従って音声入力を利用できる。

【0100】また、ある特定の応用プログラム7に他の全ての応用プログラム7のタスクの管理(音声認識結果を送るか否か、どの認識対象語彙により音声認識を行うか否かなどの処理)を行わせることもできる。

【0101】図22は、ワークステーションなどのマルチウィンドウ環境で、音声により操作できるメールツール、シェルツール、テキストエディタおよびタスク管理プログラムを示している。ここでは、どれか1つの応用プログラム7が音声入力を可能としている。この場合、テキストエディタが音声入力対象となっている(タイトルの色の変更によりそれが表示されている。)。そして、音声入力対象となっていることは、タスク管理プログラムでも同様に表示することができる。この例では、音声入力対象の変更は、タスク管理プログラムの表示の上をマウスなどのポインティングデバイスを利用して指定することができる。

【0102】(第6実施例) 実施例5では、1つの応用プログラム7だけを音声入力対象としたが、複数の応用プログラム7を同時に認識対象することも可能である。

【0103】図20の音声認識システム6に、例えば図23に示すような応用プログラム管理テーブルを持たせる。この応用プログラム管理テーブルは、音声認識システム6に接続している全ての応用プログラム7に関して認識の可否および認識対象語彙に関する情報を持つ。

【0104】このテーブルの情報の変更は、各応用プログラム7のメッセージ入出力部71からの要求によって行う。図23では、メールツールとシェルツールが音声入力可能になっている。図23の状態は、例えば図24に示すように表現できる。

【0105】ここで音声認識システム6は、「プロセス」「ホーム」といった音声入力は、シェルツールに送

り、「先頭」「次」といった音声入力、メールツールに送るといったように、認識した結果を自動的に振り分けることができる。また、「終了」は、メールツールとシェルツールに同時に送ることができるため、各応用プログラム7はそれを受け取って応用プログラム7自身を終了させることができる。

【0106】さらに、複数の応用プログラム7を音声入力対象とすることを前提とすれば、次のような操作が可能になる。図25は、タスク管理プログラムの機能を拡張した例である。「排他制御」は従来あるように音声入力対象の応用プログラム7を常に一つにする機能である。「全部」は、音声認識システム6に接続されている全ての応用プログラム7を音声入力対象とする機能である。「反転」は音声入力対象を逆転させる機能であり、メールツールとシェルツールが音声入力対象となった状態で「反転」することで音声入力対象がエキストエディタとなる。もう一度「反転」すれば元に戻る。これらの操作は、マウスのようなポインティングデバイスだけでなく、音声やキーなどの入力装置により可能である。例えば、何等かのキーボタンやキーを押しながら音声入力する。

【0107】「全部」ボタンを押しながら発声すると、全部の応用プログラム7が音声入力対象となり、「反転」ボタンを押しながら発声すると、音声入力対象が反転し、ボタンを離すとそれらの状態は元に戻る。

【0108】特定の1つの対象を指定しないまま入力し、その入力が適切に処理されることが本実施例では可能になる。ワークステーションなどのマルチウインドウ環境を考えると、その上で例えば音声による操作が可能な応用プログラム7が複数動作していたとしても、対計算機ということと考えれば、人間の相手は1つであり、計算機も相手の発声をタスク切換えなどの特別な操作をすることなく、自動的に適切に処理されることを人間が期待するのは自然のことであり、音声メディアの特性を活かすことになるといえる。

【0109】(第7実施例) 上述の第6実施例において、各応用プログラム7の認識対象語彙が何であるかは分からない。そのため、タスク管理プログラム(あるいは別の応用プログラム7にしてもよい)に各応用プログラム7の認識対象語彙を表示させる。応用プログラム7は音声認識システム6に対して、音声認識システム6の持つ応用プログラム管理テーブル(図23)の情報を要求することで、その表示が可能である(図26)。

【0110】このように音声入力対象となった応用プログラム7の認識対象語彙を自動的に表示することで、ユーザが各応用プログラム7ごとに入力に使用する認識対象語彙を記憶する必要がなくなり、ユーザの負担が少なくなる。また、応用プログラム7の作成者の側にも認識対象語彙を表示させる手段を用意する必要がなくなる分、負担解消が計れる。これはまた、例えば、入力対象

の応用プログラム7の表示と一緒に表示できる(図27)。図27では、メールツールとシェルツールの色の変化を持って、それらが入力対象となっていることが表示されている。

【0111】(第8実施例) 複数の応用プログラム7の制御は、画面の表示やマウスなどのポインティングデバイスを必ずしも必要としていない。例えば、音声によるビデオ予約が可能なVTR制御プログラムを電話でコントロールしている際に、第5実施例で述べた音声メールプログラムがVTR制御プログラムの処理に一時的に割り込み、「緊急のメール受信しました。ないようを確認しますか」と合成音声を出力して知らせることができる。この確認を受けた利用者は、受信したメールの内容を合成音声により知ることができる。

【0112】メールによる作業が終わると、ビデオ予約の作業が再開される。VTR制御プログラムは、作業中断に備えて「予約内容確認」などの語彙とともに、中断前までに行われた予約内容を確認できれば、より使いやすいインターフェースとなる。電話の場合、音声だけでなく、電話のプッシュボタンなどの入力装置が利用できる。音声入力の自然な性質を活かしながら、例えば環境の雑音が一時的に増大し、音声による入力が疎外されるような場合には、適宜プッシュボタンなどを利用して入力を確実にすることができる。

【0113】(第9実施例) 次に、本発明による音声認識プログラムによる認識語彙の学習に関する実施例について説明する。

【0114】従来、認識語彙の学習の際は、学習語彙の一覧表の中から利用者が学習させたい語彙を選択するが、語彙が多い場合、選択したい語彙を探すのに手間がかかり、使い勝手を悪化させていた。例えばワークステーション用に発売されている音声認識装置における学習プログラムでは、様々な応用プログラムで使用する認識語彙がすべて表示されるため学習させたい語彙を数百の単語リストから選ばなければならなかった。

【0115】本実施例では、応用プログラムからの認識語彙情報を利用することにより、利用者に提示する単語一覧の語彙数を少なくして容易に目的の語彙を選択するようにでき、また、応用プログラム使用中であってもその場で学習を行うことができるようにしている。

【0116】同実施例は、図28に示すように図1で述べた音声認識システム1と応用プログラム2に学習データ収集部8と辞書作成部9を加えた構成からなっている。

【0117】ここで、学習データ収集部8は、音声認識システム1とメッセージ交換を行って応用プログラム2に関する語彙情報を受け取り、利用者への語彙表示を行って認識語彙を選択させる。また、学習に必要な設定、例えば、学習データの出力を行うように音声認識システム1へ要求し、受け取ったデータをファイルに保存す

る。辞書作成部 9 は上記ファイルを入力として認識辞書の作成を行う。

【0118】以上の動作を行うために、学習データ収集部 8 は、図 29 に示すように単語音声特徴データ保存部 81、学習語彙表示選択部 82、学習データ収集制御部 83、学習語彙ガイド表示部 84 から構成している。

【0119】ここで、学習語彙表示選択部 82 は、語彙を利用者に表示して学習語彙を選択させるもので、内部に有する学習語彙テーブル 821 に音声認識システム 1 から送られて来る応用プログラム 2 の認識語彙を記憶するようになっている。学習語彙テーブル 821 は、例えば文書編集に使うコマンド群が認識対象になっている場合は、

音声認識対象語彙：取り消し、カット、コピー、ペースト、フォント

のようになっていて、この内容が、例えば図 33 のように表示され、利用者が応用プログラムを使用しているその場で目的の語彙を選択することができる。表示される語彙は応用プログラムの内部状態に応じて必要とされる認識対象の語彙のみであるため、全部をまとめて表示するよりも非常に少なくすることができ、容易に目的の語彙を選択可能である。単語音声特徴データ保存部 81 は、メッセージ処理部を介して音声認識システム 1 から送られてくる単語音声特徴データを、例えば磁気ディスクなどに保存する。学習データ収集制御部 83 は、データ収集の全体制御を行い、データ収集の開始/終了を示すためのデータ収集指示フラグを持つ。音声認識システム 1 との間のメッセージ交換は、図 30 に示すメッセージを用いて行うことができる。

【0120】学習データ収集のため、音声認識システム 1 では、音声認識を行って認識結果を応用プログラム 2 へ送る通常の認識動作の他、音声分析の結果得られる単語音声特徴データをデータ収集部 8 へ返すデータ収集動作の 2 つの動作モードを行うことが可能であり、以下では各々の動作を認識モード、学習モードと呼ぶことにする。

【0121】次に、図 31、図 32 を参照しながらデータ収集の手順について説明する。

【0122】図 31 は、音声認識システム 1 のデータ収集時のフローチャートである。

【0123】この場合、学習を行う前に音声認識システムでは、応用プログラムとの通信により、すでに認識語彙が設定されているものとする（ステップ 3101）。そして、データ収集部 8 からの学習モード設定要求メッセージを受信すると（ステップ 3102）、学習に必要な動作を行う（ステップ 3103）。

【0124】学習に必要な動作は、例えば設定されている語彙のセットをデータ収集中に保持するために音声フォーカスを移らないようにしたり、収集中に認識結果を応用プログラムへ送って認識結果によって応用プログラム

2 の状態が変化して設定語彙が変化しないようにデータ収集中、認識結果を応用プログラム 2 へ送らないようにすることなどがある。

【0125】次に、音声認識システム 1 は、データ収集部 8 へ認識対象語彙のリストを送信した後（ステップ 3104）、データ収集部 8 からのメッセージを受信し（ステップ 3105）、それが音声特徴データ送信要求であれば、音声入力がおこなわれる度に特徴データをデータ収集部 8 へ送信し（ステップ 3107）、学習モード解除要求であれば、学習モードを解除を行い通常の認識モードに戻る（ステップ 3108）。

【0126】図 32 は、学習データ収集部のフローチャートである。

【0127】まず、初期状態としてデータ収集の実行を指示するフラグに OFF が設定されている（ステップ 3200）。ユーザによりデータ収集が ON にセットされると音声認識システム 1 へ学習モード設定要求のメッセージを送る（ステップ 3201）。次に音声認識システム 1 にその時の認識対象語彙を要求し、語彙を学習語彙表示選択部 82 の学習語彙テーブル 821 に記憶させる。

【0128】学習語彙ガイド表示部 84 は、例えば図 33 のように表示し（ステップ 3202）、学習語彙をマウスなどを用いて選択させる（ステップ 3203）。選択語彙は複数でもよく、例えば選択した語彙の背景色が白から緑に変化して見易くすることができる。図 33 は、文書編集メニューの語彙の中から「コピー」と「ペースト」を学習語彙として選択した場合を図示している。

【0129】次に、単語音声特徴データ送信要求を音声認識システム 1 に出した後（ステップ 3204）、学習語彙の発声を促すための発声すべき語彙の表示が学習ガイド表示部 84 によって図 34 のようになされる（ステップ 3205）。この場合、ガイドはなくすることも可能である。また、補助情報として発声回数などを表示したり、発声すべき語彙を合成音声によって聞かせることもできる。こうすることで、ガイドを画面に表示するだけに比べて見誤りなどによる間違った発声を少なくすることができる。

【0130】ユーザが発声した後、音声認識システム 1 から送られてくる単語音声特徴データをファイルへ出力し、学習データ収集制御部 83 により設定されているデータ収集指示フラグによりデータ収集の送信/終了を判断する（ステップ 3207）。フラグが ON なら単語音声特徴データ送信要求から、ステップ 3209 を介して上記データ収集・ファイル出力までを繰り返し、OFF なら音声認識システム 1 に学習設定解除の要求を出す（ステップ 3208）。

【0131】次に、データ収集時の音声認識インターフェース全体の処理の流れを図 35 を用いて説明する。

【0132】まず、初期設定では、ユーザからデータ収集の指示が出されると(a)、データ収集部8より音声認識システム1に対して学習モード設定要求が出される(b)。これを受けて音声認識システム1が現在認識に用いている認識対象語彙をデータ収集部8に送る(c)。

【0133】データ収集部8では、認識対象語彙をユーザに表示して学習を行う語彙の選択を促す。学習の語彙が選択されると(d)、データ収集部8は、音声認識システム1に対して単語音声特徴データの送信を要求し(f)、選択された語彙を発声のガイドとして表示し(e)、ユーザに発声を促す。

【0134】音声認識システム1では、発声されたユーザの音声进行处理した後、データ収集部8に単語音声特徴データを送信し(g)、データ収集部8は、そのデータをファイルに出力する。

【0135】学習終了時には、まず、ユーザがデータ収集終了の指示を入力し(h)、データ収集部8は、学習モードの解除を音声認識システム1に要求する(i)。音声認識システム1では、それを受けて学習モードを解除する。

【0136】データ収集終了後は利用者が必要に応じて認識辞書の作成を行うことができる。辞書作成部9は、単語音声特徴データ保存部81からのデータを用いて辞書作成を行い辞書をファイル出力する。

【0137】従って、このようにすれば目的の語彙を簡単に選択でき、応用プログラム使用中においても認識語彙の学習を簡単に行うことができるようになる。

【0138】(第10実施例)次に、時間のかかる辞書作成をバックグラウンドで行い、データ収集中や他の応用プログラム実行中に辞書を作成することで辞書作成終了を待たずに使い勝手の良い音声認識インターフェースを実現する実施例について説明する。

【0139】ところで、従来、音声認識のパターンマッチング法としては、DP法やHMM、複合類似度法などが知られており、いずれも標準となる認識辞書を用いてパターンマッチングを行うが、例えば高精度な認識を行うため固有値展開などを必要とする複合類似度法(永田、他“ワークステーションにおける音声認識機能の開発”電子情報通信学会技術報告、HC9119、pp. 63-70、(1991))では、辞書作成のための計算量が多く、現在高速であるとされるワークステーション、例えば処理能力20MIPSの計算機を用いた場合でもかなりの時間、例えば一単語当り数秒から数十秒を要するため待ち時間による学習インターフェースの使い勝手の悪化が無視できない。そこで、学習データの収集中に辞書作成をバックグラウンドで計算することにより、待ち時間を減らしてインターフェースの使い勝手を向上させるようにしている。

【0140】そこで、同実施例では、辞書作成をバック

グラウンドで行うことでインターフェースを良くする音声認識システムについて説明する。

【0141】この場合、図28で述べた辞書作成部9を、図36に示すように辞書作成管理部91、辞書作成制御部92、データ入力部93、辞書作成部本体94、ファイル出力部95から構成している。

【0142】ここで、辞書作成管理部91は、データ収集部8からのメッセージを受け、要求された語彙の単語認識辞書の作成を辞書作成制御部92へ指示し作成終了をメッセージでデータ収集部8に通知する。

【0143】複数の辞書作成要求があった場合に順序よく実行するため例えば図37のような辞書作成管理テーブルの要求日時の順番に従って作成を行う。図37は例として文書編集用のコマンドである「コピー」「ペースト」「カット」という単語について、この順序で辞書作成を要求されたときの管理テーブルの内容である。語彙などの条件は要求のあった日付、時刻とともに管理テーブルに登録され、辞書作成がこの順で行われ、作成の終了した要求は管理表から削除される。

【0144】辞書作成要求は上記のように語彙を指定するだけでなく、単語音声特徴データの属性としてデータ自身に登録されている他の情報、例えば図38のように発声者の名前を指定してその人の特定話者用の辞書を作ったり、図39のように日付を指定して新しいデータのみによって辞書を作ることもできる。

【0145】そして、辞書作成管理部91と辞書作成制御部92の間はメッセージ交換でやりとりを行う。

【0146】次に、図40、図41を用いて辞書作成の流れについて説明する。

【0147】まず、図40は辞書作成管理テーブルへの登録の手順である。この場合、辞書作成要求のメッセージがあったかどうかを判断し(ステップ4001)、なければ要求を待ち、あれば語彙やユーザ名などの条件を辞書作成管理テーブルに登録する(ステップ4002)。

【0148】一方、図41は辞書作成の手順である。この場合、辞書作成管理テーブル上に登録されている辞書作成要求を検索し、要求がなければ登録を待ち、あれば最も古い日時の要求を選ぶ(ステップ4101)。次に単語音声特徴データを入力し(ステップ4102)、上記要求の条件に適合するデータを選択する(ステップ4103)。選択したデータのみを用いて辞書を作成しファイル出力する(ステップ4104、4105)。上記要求を管理テーブルから削除し、管理テーブルの検索(ステップ4101)へ戻る。以上を繰り返す。また、すべての辞書作成要求が削除された時点で、辞書作成が終了したことを学習データ収集部に通知しても良い。

【0149】認識辞書の作成は、データ収集時にバックグラウンドで行うため、辞書作成の進行状況は利用者にとって分かりにくい。そこで、辞書作成の進行状況を例

えば図42(a)(b)に示すように全処理量に対する終了した処理量の割合を表示することによって利用者に分かりやすいインターフェースを提供できるようにしている。この場合、辞書作成の開始や終了の際には、ビープ音などにより通知することも可能である。また、辞書作成処理の速度を表示することも可能で、例えば図43に示すように速度を4段階に分けたり、図44(b)に示す色分けを用いて同図(a)のように色で処理速度を表示したりでき、計算機の負荷が大きくて辞書作成の処理が進まない場合には、処理が停滞していることを表示することにより、利用者に計算機の負荷の分散を促すようにもできる。

【0150】以上のように、時間がかかる音声データの収集中にバックグラウンドで辞書作成を行うことにより、待ち時間を少なくして使い勝手の良いインターフェースを実現することができる。

【0151】また、以上述べた辞書作成は、独立したプロセスとして動作することが可能で、データ収集部8からの要求だけでなく、音声認識システムやその他の応用プログラムからも辞書作成要求を受け付けることが可能であり、学習データ収集処理時のみに限らず、いつ辞書作成を行ってもよい。

【0152】(第11実施例) 認識対象を単語または文節などとする音声認識においては、従来より入力音声のパワーの変化、音声ピッチの変化、あるいは零交差回数などの特徴パラメータを用いて単語境界を検出し、この音声特徴ベクトルと認識語彙セットについての認識辞書とを照合することにより行われていた。しかし、実際の作業環境では、背景雑音やユーザの不用意な発話(他のユーザとの会話や独り言など)の影響により誤った単語境界が検出されることが少なくない。このため、音声認識システムのユーザは現在何が認識対象になっているかを常に意識し、それ以外の言葉を発声しないようにする必要がある。

【0153】一方、音声を計算機への入力手段の一つとして他の入力手段(例えばキーボードやマウス)と合わせて作業を行う場合、ユーザは、入力内容や作業の状況に応じてそれぞれの入力手段を使い分けことが考えられる。

【0154】そこで、本実施例では、図45に示すように図1で述べた音声認識システム1と応用プログラム2に音声認識自動停止部10を加えた構成とし、認識処理に、通常の認識処理(現在の認識対象となっている全ての語彙に対する認識処理)をおこなうモードと、特定のキーワードについてのみ認識処理を行うモードの2つのモードを設け、認識処理を開始して暫くは通常の認識処理を行い、予め定めておいた時間内に音声入力が行われなかった場合には、それまでの認識語彙セットを保存し、特定のキーワード(例えば「認識開始」など)のみを認識語彙セットとするモードに切り替わるようにす

る。その後、このキーワードが入力されれば、保存していた認識語彙セットを新たに設定し、通常の認識処理モードに移行する。この認識処理モードの切り替えは、例えば音声フォーカスの変更や音声以外の入力手段による指示によっても行われ、認識モードの移行は、メッセージまたはアイコンによる表示やビープ音などを用いてユーザに伝えられる。これにより、ユーザが音声を暫く使わない状態になると、自動的に音声認識のモードが切り替わり、特定のキーワード以外の音声を無視することで検出誤りによる予期しないタスクの切り替えや誤動作を回避することができる。

【0155】また、ユーザはキーワードを発声するか、音声以外の入力手段により音声認識処理モードの切り替えを意識的に行うことができる。上記の処理は、例えばインターバル・タイマ機構を用いることにより実現できる。これは、現在時刻から時間切れになる時間を秒数で指定するもので、時間切れになると、その旨を通知するシグナルが渡される。このシグナルを受信した時点で音声認識のモードの切り替えを行う。

【0156】以下、図46に示すフローチャートに従って説明する。

【0157】まず、最初にタイマが時間切れになるまでの秒数を設定し(ステップ4601)、時間切れか否かを示すフラグを0にする。このフラグは、時間切れになった旨を通知するシグナルを受信した際に呼び出されるシグナルハンドラ内で1がセットされるようにしておき、認識処理の最初にその値が調べられる。なお、タイマの機能は、計算機に通常内蔵されている時計の機能により容易に実現可能である。また、シグナルハンドラは、音声認識自動停止部10の中にプログラムとして書くことができる。

【0158】次に、認識対象とする語彙セットを設定した後(ステップ4602)、時間切れか否かを調べて(ステップ4603)、時間切れでなければ、その語彙セットに対する認識処理を行う。

【0159】認識処理は、まず入力音声のパワー変化や音声ピッチの変化、あるいは零交差回数などの特徴パラメータを用いて音声区間の始端と終端を検出し(ステップ4604)、終端が検出されれば、その始端と終端で定まる音声区間から音声特徴ベクトルを抽出し、現在の認識語彙セットの認識辞書と照合を行い、各確認語彙の類似度を求め、そのうち類似度最大で、かつその値が予め定めておいたしきい値以上のものを認識結果として出力し、認識処理を終了する。(ステップ4605~4609)

なお、図46では、音声特徴ベクトルの抽出から、認識辞書との照合およびしきい値による判定までを認識処理としている。終端が検出されない場合や、認識結果が得られない場合は(ステップ4605、4607)、語彙セットの設定に戻り、必要に応じて(例えばクライアン

トから音声フォーカスの変更や認識語彙の変更要求があった場合)認識語彙セットの変更を行い、時間切れか否かを調べて、時間切れでなければ再び現在の認識語彙セットに対する認識処理を行う。時間切れになった場合は、それまでの認識語彙セットを保存し、特定のキーワードを認識語彙とするモードに移行する。そのキーワードが検出されるか、クライアントから認識処理モードの切り替え指示があれば、保存していた認識語彙セットを復元し、タイマを再設定して通常の認識処理に復帰する(ステップ4610~4617)。

【0160】以上述べた認識機能の自動停止機能により背景雑音やユーザの不用意な発話による誤動作を防ぎ、使い勝手のよい音声認識インターフェースを実現することができる。

【0161】また、背景雑音やユーザの発話による誤動作をユーザが意識的に避ける方法として、従来からマウスやキーを押し下げている間だけ、音声入力を行う方法が使われているが、音声入力ごとに毎回マウスを操作するのは煩わしいという問題がある。そこで、常時音声入力中として、マウスを押し下げている間だけ音声入力を受け付けずにすれば、発声ごとにマウスを操作しなくてもいいという煩わしさを軽減できる。

【0162】(第12実施例)ところで、音声メールツールは、音声入力可能な電子メールシステムであり、音声を使って受信したメールのリストを移動して内容を確認したり、そのメールに対する返事を送信することができる。

【0163】この場合、ツールは、リスト表示部、受信メール表示部、送信メール編集部からなり、リスト中の反転表示されたメールが受信メール表示部に表示される。そして、例えば、音声を使って以下のような操作ができる。ここでは、上司からの緊急のメールに対して返事を出すまでを示している。

【0164】「メールツール」(音声メールツールを全てウインドウの前に出す。)

「先頭」(受付けリストの先頭にリストポインタを移動する。)

「次」(リストポインタを次のメールに移動する。)

「最後」(受信リストの最後にリストポインタを移動する。)

「前」(リストポインタの前のメールに移動する。)

「上司」(上司からのメールだけをリストアップする。)

「緊急」(そのうち緊急のメールだけをリストアップする。)

「返事」(緊急のメールに対して返事を出す。送信メール表示部に「To: 上司名」と「Subject: Re: 上司からのメールのSubject」が入

る。)

メールシステムの初期状態を図47に示す。メールリストの表示部には、全てのメールリストを一度に表示できないため、所望のメールを探すのにマウスを使う場合には、表示部の右側にあるスライド用のバーを使う必要がある。特に大量のメールが来た時などは、メール探しに多くの労力を必要とし、操作性は十分であるといえない。しかし、ここで音声を用いることにより、直接所望のメールを検索でき、作業の大幅な効率化が図れる。

10 【0165】ここで、例えば上司からの緊急のメールを選択する場合、「上司」「緊急」と発声するだけで、選択することができる。図48に上司からの緊急のメールの検索結果を示す。この例では2通のメールがきているものとすると、次のようになる。

【0166】「コピー」(メッセージをコピーする。)

「ペースト」(コピーしたメッセージを受信メールにペーストする。)

「引用」(そのメッセージに引用符を付ける。)

20 ここで、そのメッセージに対する返事を書き、

「サイン」(必要があれば自分のシグネチャをメールの最後に付ける。)

「送信」(返信メールを送信する。)

ここで使われている「上司」や「緊急」は、音声マクロコマンドとして実装されており、メールのヘッダや内容を用いて照合した結果を用いてリストを限定するものである。すなわち、電子メールの発信者の名前、所属、標題、差出日、本文の内容は、テキスト(文字データ)で書かれており、その内容を理解し、キーワードや内容の照合を行うことにより、音声での効率的な電子メールの取り出しが可能になる。これはフルテキストサーチなどの情報検索技術や文脈解析技術を用いて、WS上で実現でき、音声入力インターフェースの利用により音声メールの使い勝手が大幅に向上する。また、テキストの一部を音声合成で読み上げたり、強調したり、スピードを変化させることも可能である。また、図47に示すように認識語彙の表示や現在音声フォーカスが当たっているクライアントの表示、認識が動作中であるか否かの表示などを行い、ユーザにシステムの状態をできるだけ伝えるように考慮し、作業の効率化を可能にしている。

30 【0167】(第13実施例)音声認識サーバを使って、既存のアプリケーションを音声で制御することができる。これは、既存のアプリケーションのキーボード入力を音声によって代行するクライアントを作成すれば可能である。ここでは、既存のアプリケーションに対する音声制御を可能とする音声マクロプログラムを使って、既存のDTP(Desk Top Publishing)システムを音声コントロールする例を示す。

50 【0168】音声マクロプログラムは、既存アプリケーションの認識語彙に関する知識をメニュー形式で持ち、

そのメニュー階層を利用して認識語彙を限定する。ここで、

“図形”メニュー
 “取り消し”
 “グループ化”
 “グループ解除”
 “フロント”
 “バック”
 “上／下（うえした）反転”
 “右／左（みぎひだり）反転”
 “回転”
 “トップレベル”メニュー
 “文書”
 “編集”
 “図形”

メニュー階層のルートを「トップレベル」と呼び、トップレベルから単語を発生し、メニュー階層をたどることでコマンドを実行していく。メニューの階層を移動するごとにウインドウにメニューの各項目とメニュー階層における現在位置をパスの形で表現しユーザに呈示する。

【0169】そして、以下のように操作される。ここでは、文書ウインドウに存在する複数の図形を取り扱う例を示している（図49参照）。

【0170】図形を扱うためにトップレベルから図面メニューを開く。

「図形」（メニューの項目が音声コマンドにリストアップされる。）

ここで、文書ウインドウ上の複数の図形をマウスで選択する。

「グループ化」（複数の図形を1つの図形として取り扱うべく纏める。）

「上下反転」（グループ化した図形の上下を反転させる。）

「回転」（図形を回転させる。）

「グループ解除」（グループ化を解除する。）

次に、先にグループ化された図形のうちの1つをマウスで選択する。

「バック」（選択した図形を全ての図形の後ろに送る。）

「取消し」（「バック」により行われた操作を取り消す。）

「フロント」（一番前に送る。）

これをマウスを使って操作する場合には、

・メニューバーをクリックしてメニューを表示する。

【0171】・メニューをプルダウンし、実行したいコマンドの項目を選択する。

【0172】・マウスボタンから手を離してコマンドを実行。

の少なくとも3アクション必要であり、マウスポインタの移動の手間を考えると、それ以上のアクションを行っ

ていると考えられる。

【0173】ところが、音声を使用すると、

・操作を行う単語を発生する。

の1アクションで済むため、音声の有用性分かる。マウスを使ってメニューを選択することで操作する場合には、例え予め何を操作したいかをユーザが分かっている、上記の操作は必ず実行しなければならない。音声は、他の入力手段と組み合わせることで、より効果的なインタフェースとなる。

10 【0174】ここで、キーボードマクロを使えば、音声と同様に1回だけの操作で済むが、キーボードマクロは基本的に一つの文字で表現するため、キーボードマクロが多ければ多いほど対応付けのしにくい文字とコマンドの組み合わせを記憶することが要求され、ユーザの負担になる。

【0175】そこで、コマンドを、ただ1つの文字でなく、そのコマンドの意味をも自然に表現し得る音声と結び付けることで、アプリケーションは、ユーザに対して、より自然なインターフェースが提供できる。

20 【0176】また、単語認識の際に上述した図形メニューのなかで、例えば「グループ化」と「グループ解除」のように前半部分が同じカテゴリに存在する場合には、部分抽象化により単語の後半部分のパターンを用いて認識を行うことにより、認識精度の向上を図ることができる。また、「上下反転」「左右反転」のように後半部分が同じ場合には、単独の前半部分のパターンを用いて認識を行うことも可能である。要するに、パターンの違いがより明確になるように様々な視点から認識のための単語パターンを取り出し、認識を行うことにより認識性能の向上が可能になる。

【0177】（第14実施例）以上、述べてきた音声認識インターフェースは、音声の入力にのみ注目してきたが、音声の出力機能をインターフェース内に取り入れ、テキストからの音声合成や音声データの再生を行なうようにすれば、音声の入出力を統合して行なうことができるため、複数の応用プログラムへの音声入力とそれらからの音によるメッセージの出力を簡単に行なうことができ、ユーザにとって取扱い易いインターフェースを実現することができる。

40 【0178】以下に、音声合成機能を備えた音声認識インターフェースである音声入出力インターフェースの構成について説明する。

【0179】図50は音声合成部を備えた音声入出力システムの概略構成を示しており、図1で述べた音声認識システム1に音声合成部14を付加した構成になっている。この場合、音声合成部14はメッセージ処理部11からの指示に従ってテキスト情報から合成音声生成を行い、音声出力を行なうようになっている。また、応用プログラム管理テーブル13は、複数の応用プログラム2からの音声出力を制御するため、図55に示すように応

用プログラム2の音声出力に関する情報を収納するフィールドを持っている。これにより、複数の応用プログラム2からの音声出力に対する制御を行なうことができる。ここでの音声出力に関する情報としては、特定の音声出力に対して音声出力を優先的に行なうことを指示するための音声出力優先度などがある。

【0180】図51は、音声合成部14の概略構成を示しており、全体制御部561、波形重畳部562、音声出力管理テーブル563、波形合成部564からなっている。

【0181】全体制御部561はメッセージ処理部11から合成音声の出力要求とともに文字列を受けとり波形合成部564に送って音声合成を行ない音声出力する。この場合、音声合成部14によって出力する音響信号は合成音のみでなく、録音された音声や音声以外であってもよく、その場合は音声の合成を必要としない。このときは波形合成は行わずにメッセージ処理部から受けとった波形データをそのまま音声出力するようにしている。

【0182】また、波形合成部564は全体制御部561から文字列データを受けとって音声合成を行なう。音声合成の方式としてはさまざまな方法が知られており、例えば文献(D.Klatt: "Review of text-to-speech conversion for English", J.Acoust.Soc.Am., 82, 3, pp.737-793 (Sept.1987))の方法を用いることが可能である。

【0183】音声出力管理テーブル563はメッセージ処理部11からの音声出力の要求を登録するテーブルであり、このテーブルに登録された順番に従って音声出力を行なうことにより、複数の音声出力要求に対して時間的な整合性を保ちながら音声出力を行なうことができる。

【0184】音声合成部14は独立したプロセスとして動作させることが可能で、メッセージ処理部11とは、音声認識システム1と応用プログラム2の間のメッセージで述べたように、プロセス通信によるメッセージ交換によりデータのやりとりを行なう。ここでのメッセージとしては図53に示すようなものがある。

【0185】同図(a)の応用プログラム2からメッセージ処理部11へのメッセージは応用プログラム2からの命令を意味している。ここでの音声合成要求は、応用プログラムがテキスト内容を合成音声に変換させる要求で、合成するテキストデータと共に要求を出し、その結果合成音声データが通知される。波形再生要求は応用プログラムが録音等により既に波形の形で音声データを持っている際、それをそのまま再生するための要求で、再生データと共に送信する。音声合成・再生要求は、音声の合成とその再生をまとめて行なう要求であり、合成音声データは通知されない。

【0186】優先度設定要求は、特定の応用プログラム

からの出力音を優先させるための要求であり、例えば出力音のレベルと音声合成処理の優先度、中断出力の有無、などに関して、設定できるようになっている。

【0187】音声出力要求の優先度は、例えば緊急を要する場合に、高い値に設定することにより、直ちにユーザの注意を向けることができるため効果的である。

【0188】先に述べたように、音声出力管理テーブル563はメッセージ処理部11からの音声出力要求を登録するテーブルであり、このテーブルに登録された順番に従って音声出力を行なうことにより、複数の音声出力要求に対して時間的な整合性を保ちながら、音声出力を行なうことができる。

【0189】音声出力管理テーブル563の例を図52(a)(b)に示している。テーブルに登録するデータはデータID、波形かテキストかを表す入力データの種別、出力要求のテーブルへの登録時刻、テキストデータの内容、音声出力の際の音量などがある。図の例では、データID#1、#2、#3がテキストデータであり、#0~2のデータに対しては処理が終了しているが、#3のデータは現在処理中、#4のデータはまだ処理が行われていないことを示している。

【0190】一方、メッセージ処理部11から応用プログラム2へのメッセージは図53の(b)に示すような種類がある。音声出力状況通知は、要求された音声出力が終了したことを通知し、優先度設定通知は、優先度設定要求に従って音声出力の優先度が設定されたことを通知する。いずれも要求に対する確認のメッセージである。

【0191】応用プログラム2がどのメッセージを受け取るかの設定は、先の音声認識システム1と応用プログラム2の間のメッセージに関する説明で既に述べた通りで、入力マスクによって設定することができる。この場合、音声合成部14が加わったことにより、図54に示すような種類からなっている。

【0192】また、上述したようなメッセージ以外にも、エラーメッセージや音声出力レベルの設定メッセージ、音声合成部14の内部情報にアクセスするメッセージなどさまざまなメッセージが設定可能である。

【0193】音声合成部14とメッセージ処理部11との間もメッセージによって情報交換が行なわれる。この場合のメッセージは図53の(c)(d)に示す種類がある。このうちの(d)のメッセージ処理部11から音声合成部14へのメッセージは、(a)の応用プログラム2からメッセージ処理部11への要求メッセージとほぼ同じであり、(c)の音声合成部14からメッセージ処理部11へのメッセージは、(b)のメッセージ処理部11から応用プログラム2への通知メッセージとほぼ同じ種類のものを使うようにしている。

【0194】以上、述べたように音声合成部14を有する音声認識システム1の各部においてメッセージをやり

とりすることによって、複数の応用プログラム2からの要求による音声出力処理が進められるが、次に、音声認識インターフェース全体としての処理の流れを図56、57に従って説明する。

【0195】図56では、既に第1実施例で述べた手続に従って応用プログラム2と音声認識システム1との接続処理と音声認識に関する初期設定をステップ6101で既に完了しているものとする。そして、ステップ6101の終了後、応用プログラム2は音声出力処理に関する初期設定を後述の図57の(a)に従って行なう(ステップ6102)。初期設定としては、音声合成部14における音声出力管理テーブル563の初期化、応用プログラム管理テーブル13の音声出力優先度情報の初期化などがある。そして、音声入力および音声出力の処理を実行する(ステップ6103)。

【0196】次に、応用プログラム2からの音声出力に関する要求ごとの音声出力処理について説明する。

【0197】まず、図57の(b-1)の音声合成要求が応用プログラム2から出された場合、メッセージ処理部11は要求をそのまま音声合成部14へ音声合成要求として送る。そして、音声合成部14は音声出力管理テーブル563へのメッセージの登録を行なう。音声合成要求は波形の再生処理を含まないため、例えば図52の出力管理テーブルメッセージID#1のように、出力ありなしの項は出力なし(=0)となる。この場合、音声出力優先度情報は使われない。合成処理が終了後は、音声合成部14は終了したことを音声出力状況通知によってメッセージ処理部11へ通知し、メッセージ処理部11はそれを応用プログラム2へ通知する。応用プログラム2はこの通知の後音声波形データ要求を出し、合成音声ごとに受け取る。

【0198】次に、図57の(b-2)の波形再生要求があった場合、メッセージ処理部は図55に示す応用プログラム管理テーブルに登録してある優先度情報を検索し、要求を行なった応用プログラムに関する情報を付加して音声合成部14へ波形再生要求を行なう。

【0199】音声合成部14では、音声出力管理テーブルにメッセージの登録を行なうが、この場合は、例えば図52のメッセージID#0または#4のような内容が登録される。波形再生終了後に音声合成部14は、音声出力状況通知により、再生が終了したことをメッセージ処理部11に送り、メッセージ処理部11はそれを応用プログラム2へ送る。

【0200】次に、図57の(b-3)の音声合成再生要求があった場合は、波形再生の場合と同様な処理で音声の合成および再生の処理を行なう。

【0201】また、図57の(b-4)の優先度設定要求によって音声出力優先度を変更することができる。音声出力優先度は先に述べたように、音声出力のレベル、音声合成処理の優先度、中断処理の有無等がある。出力

音声のレベルを高くすれば、その出力メッセージに対する注意を引きつけるのに役立ち、音声合成処理の優先度を高くすれば、その音声データが音声合成後出力されるまでの時間遅れを小さくできる。又、中断処理は、特定の音声出力データ以外の音声出力を一時中断し、そのデータのみを出力する処理であり、これらを組み合わせて使用することにより、重要なメッセージを優先的に出力するなどの処理が可能である。

【0202】例えば図52では、メッセージID#0の波形再生要求に対しては、出力レベル=3、中断出力なし、合成処理優先度(値なし)が設定されている。この場合、優先度の値は0~10の範囲で設定するようになっており、出力レベル3は、比較的小さい値である。又、中断出力なしのため、この波形データは他の音と重なって聞こえて来る。これに対し、#2の音声合成・再生要求に対しては、出力レベルは最大の10であり、かつ音声合成処理の優先度も最大であるため、合成音データが直ちに出力される。又、中断出力ありのため、この間に他の音は出力中断状態にある。この合成音を出力中は、他の音に邪魔されずに音を聞くことができる。

【0203】次に、以上述べたような音声出力要求を順次処理する方法について説明する。

【0204】複数の音声出力要求は音声合成部14の音声出力管理テーブル563に従って処理を行なう。音声出力管理テーブル563には要求のあった順番に要求のID、入力データの種別(波形/テキスト)、要求受付時刻、データ内容、処理状態、音量、出力中断処理のあり/なし、音声合成処理の優先度、排他処理の係数、等が登録される。

【0205】図58に示すように、まず、全体制御部561は、音声出力管理テーブル563の処理状態の項を参照し(ステップ6301)、「未処理」となっているデータを探し、あれば処理状態を「処理中」に更新し(ステップ6302)、データの種別を参照する(ステップ6303)。そして、データがテキストであればテキストデータを波形合成部564へ送って音声合成を行い(ステップ6304)、合成音データを波形重畳部562へ渡し、波形データであればそのまま波形データを波形重畳部へ渡す(ステップ6305)。そして処理状態を「終了」に更新して(ステップ6306)、次の未処理データの処理を行なう。

【0206】波形合成部564では、処理を行なっているデータに関する合成処理優先度情報をもとに、合成演算を行なう処理の他の処理に対する優先度を設定して演算を行なう。優先度の設定は、例えばワークステーションのオペレーティングシステムとして一般的であるUNIXのシステムコールを用い、合成プロセスに対する演算装置の割り当て時間を変更させたり、処理量の異なる複数の音声合成器を用意して優先度に応じて使う合成器を変えたりすることにより行なえる。

【0207】波形重畳部562では、波形データと共に音量、出力中断処理のあり／なし、排他処理の係数などの情報に基づいて複数の波形を重畳する。重畳の際には、時刻と波形データのサンプルの対応を常に監視し、複数の音声出力要求の間の時間とそれらの要求に対応する複数の波形データの出力される間隔が、なるべく等しくなるようにしている。また、重畳の処理は単位時間、例えば10msecごとのブロック処理によって行なうことが可能である。

【0208】次に、図59により、中断処理のある音声データを重畳する際の例を説明する。この場合、データは図52の音声出力管理テーブル563にあるデータID#1～3であり、簡単のため、登録から波形重畳までは時間遅れがないものとしたが、実際には用いる計算機の処理能力に応じて、音声合成やデータの移動による時間遅れがある。音声出力管理テーブル563に記録された時刻どおりで、かつ出力中断処理を行わずに音声データを出力する場合には、図59(a)のように、データどうしが時間的に重なっているため、緊急なメッセージであるデータ#2の音声は、先頭部がデータ#1の最後と、後半部がデータ#3の前半部と重なって出力されることになる。これに対し、出力中断処理を行なう場合の(b)では、データ#2の「緊急です」が始まる時点でデータ#1の重畳を中断し、#2の処理終了後、#1の中断された時点から残りを重畳することになる。又、データ#3は、#2が終了後に重畳される。データ#1のように、中断処理によって時間的に分割されるデータは、上述のように分割したまま出力しても良いが、中断処理後にもう一度最初から出力し直したり、又、分割された後半部は出力しない、あるいは徐々に音量を下げて重畳するなど様々な処理が考えられる。

【0209】(第15実施例)第14実施例に記述したように、音声認識システムは、音声合成部14を組み入れ、マルチタスク環境において、複数のタスクから音声認識および合成機能の利用を可能にすることで、ユーザが応用プログラム2を使用する際の使い勝手が向上する。本実施例においては、第14実施例をふまえ、具体的なシステムの応用例として、音声メールツールについて音声合成機能を追加した際の効果を中心に述べる。

【0210】図60は、第15実施例の概略構成を示しており、音声入出力システム651、ウィンドウシステム652、音声メールツール653から構成している。また、音声メールツール653は、電子メール処理部6531とメッセージ入出力部6532からなっている。

【0211】この場合、音声入出力システム651は、第14実施例に述べた、音声合成機能を持つシステムである。ウィンドウシステム652は、応用プログラムに関する情報をGUI(Graphical User Interface)を通じてユーザに提供する。そして、これら音声入出力システム651及びウィンドウシステム652を利用するこ

とで、音声メールツール653で、音声入力をマウスやキーボードと同様に扱え、音声合成をも統一的に扱えるようにしている。

【0212】通常、音声メールシステムで送受信されるデータはテキストデータであるが、テキストデータだけではなく、音声データや画像データ等をメールの中に混在させることができる。音声データを含むメールを送受信するために、メールツールは生の音声データを録音・再生する機能が必要となる。

【0213】応用プログラム2が生の音声データを扱えるようにするために、応用プログラム2と音声入出力システム651間で交わされるメッセージとして、図61に示すものを追加する。これらのメッセージを利用して、メールツールが音声データを録音する手順を図62の(a)に、再生する手順を図62の(b)に示している。また、今述べた音声の録音・再生機能を持つ音声メールツールの画面表示例を図63に示す。この表示例は、上述した第12実施例の図48とほぼ同じ表示画面を持つ。ここでは、ツールのリスト表示部の行の先頭に*印の付いたものがあるが、これは音声データを含むメール文書を識別する印である。受信メール表示部に、音声データ付メール文書の表示例を示す。メール文書中の音声データは、例えばボタン様の形式でユーザに提示する。

【0214】図63においては、緊急とラベル付けられたボタンが、音声データである。音声データをマウス等で指定し、マウスやキーや音声入力を使って再生する。音声データ付のボタンは、メールのテキスト中の任意の位置に任意の個数作成し、配置できる。

【0215】メール中の音声データの録音・再生・編集は、図64のような、音声データ編集用のサブウィンドウを用いて行なう。図の上部の2つのスライダーはそれぞれ音声データの入力・出力時のボリュームを設定するものである。その下のボタンは、それぞれ音声データの録音、再生、録音/再生の停止、音声データの編集、メールへの音声データの追加を行なうボタンである。編集ボタンには、カット、コピー、ペーストなどを行なう編集用のサブメニューが存在する。ボタン列の右端の「緊急」は、ユーザが任意に入れることのできる文字で、音声データ作成時に、ボタンのラベルとして表示される。図64の下部が音声波形データを編集する所である。データをマウスを用いて選択し、音声入力をを用いてカット、コピー、ペーストを行ったり、エコーをかけたり、ピッチを変化させたりなどの効果を音声データに加えることが可能である。また、音声データの編集やデータに対する効果の付加は、メールツールでなく、専用の音声データ編集ツールで行ってもよい。それを用いて音声を編集する際に、メールツールとの間で音声データの受け渡しを行う必要があるが、その受け渡しを音声入力を使ったカット&ペーストによって行えば音声データに対す

る編集操作が簡単に行えるようになる。

【0216】音声入力を使ったカット&ペーストは、音声データに対してだけでなく、テキストやグラフィックなど様々な形態のデータに対して適用し、応用プログラム向けのデータの受け渡しに用いることができる。

【0217】以上述べた機能を用いてメールの返事を出す場合には「返事」と発することにより読んだメールの全て、あるいは文面の一部などを自動的にコピーし、引用の印をそれぞれ付加し、さらに自分のサインと録音メッセージを自動的に付加して送信してしまうことにより、ほとんどのキーボードに手を触れずにメールの返事を出すことができる。その際、録音メッセージは、前もって録音してあるものを用いてもよいが、自動的に録音モードに入って「送信」が発声されたならば、自動的にその録音データを付加してメール送信を行ったりできる。例えば図65は、送別会のお知らせに対する返信の文面の例である。この例では、8行目まで、送られてきたお知らせのメールのコピーに引用マーク(〰)を付け、9~11行目に自分のサインと録音メッセージの印を付加するようにしている。

【0218】また、図64で示す音声データの録再・編集機能の一部または全部を図66のように、受信メール表示部や、送信メール編集部に並べて配置することで、メール中の音声データに対する操作性が向上すると考えられる。

【0219】録音データはそのまま全部をメール用のデ*

「田村殿」

「先週の出張報告書を至急提出のこと」

(緊急ボタンの音声データを再生)

「沢田」

と、このように、データの出現順に、データの種類の応じた処理(テキストデータは音声合成し、音声データはそのまま再生する)を行なうことで、テキスト以外のデータをも読み上げることができる。また、テキストデータだけの読み上げや、音声データだけの読み上げを可能にすることもユーザにとって有用になる。テキスト以外のデータ形式としては、音声以外のものでも、そのデータ形式に従った処理を行えば良い(動画なら動画の再生を行なう)。

【0226】メールの読み上げは、本文だけではなく、40 題や発信者や送受信の時間を示すメールのヘッダに関しても行なって良い。

【0227】ここで、全てのメール文書に対して、同一の読み上げ方をする必要はない。例えば、メールアドレスと、合成音声の属性を図67に示すようにデータベース化することによって、発信者毎にメール文書読み上げの際の音声の特徴を変化させることができる。図67の設定では、Tamura氏からのメールは、低くゆっくりと話す男性の声で、Nakayama氏からのメールは、高く早口の女性の声で、それ以外のメールは、標準

*ータとして用いてもよいが、データ中には言い淀みなどにより不要な無音部があり、必要以上にデータ量が多くなってしまふことがある。

【0220】そのような場合、無音部を自動的に検出して一定の長さ、例えば1秒以上の無音部をカットすることも可能である。

【0221】また、録音の際の利用者の動きにより、口とマイクロホン間の距離が変化して録音レベルが一定でなくなり、聞きづらいデータになってしまうことがある。

【0222】そのような場合、録音データのパワーを調べて全体に亘ってレベルを均一にし、聞きやすくすることができる。レベルの均一化の処理は、ある単位ごと、例えば単語、文ごとのレベルを求め、最大のレベルを持つものに他を合わせるようにするなどにより実現可能である。

【0223】また、データ全体あるいは上述の最大レベルが小さすぎたり、大きすぎたりした場合には、データ全体のレベルをそれに応じて変えることにより、聞き苦しくないようにできる。

【0224】さて、本実施例のメールツールを使うことで、テキストと音声の混在したメール文書を読み上げることができる。

【0225】図63の受信メール部のメールを読み上げることとすると、

(音声合成)

(")

(音声合成)

的な声の高さを持つ男性の声で、標準的スピードにより読み上げられる。

【0228】さらに、発信者情報だけではなく、1つの文書内の情報を使って合成部を変化させることが考えられる。例えば、引用符に囲まれた部分のみに関して、男女の性別を入れ替えるとか、声の高さや読み上げの速度を変化させることが可能である。

【0229】また、メールの受信者が、合成音声によるメールの読み上げを行なうことを想定し、メール本文中のテキストに、音声合成用の制御コードを付加して、メールの読み上げ方を指定することが考えられる。制御コード交じりのメールの例を図76に示す。

【0230】この場合、@<...>で囲まれた部分が、制御コードおよびその指定で読み上げられる部分である。male、5、5、9は、特に性別(男性)、声の高さ、速度、声の大きさを示し、ここでは、「絶対に遅れないように」の部分だけが、その他の部分よりも大きな声で読まれる。このように、メール本文中の部分に対し、音声合成の細かな設定を可能にすることで、メール中の重要な所を強調したり、文章の抑揚を変えたり、引

用した言葉を本人に近い特徴の合成音声で読ませて変化をつけるといった事が可能となる。

【0231】以上に述べたメールツールはマルチタスクの環境下で音声によって制御を行うことができるため、キーボードやマウスなどによって文書の作成やプログラム編集などを行いながら音声によってメールによって読むことができ便利である。

【0232】なお、メールツールだけでなく、情報検索のためのツール、例えば英和、和英などの電子辞書や対訳辞書、類似表現、言い換えなどを引くための類似語辞書などのデータベースを本発明によるインターフェースにより音声で操作すれば、文書やメール作成中に調べたい単語などを音声による操作で引くことができるため、文書作成の中断を少なくできて便利である。

【0233】メールの内容の確認を、表示によらず、音声読み上げを使って行なう際に、1つのメール全体を読み上げの対象とすることは、特に、大量のメールの中から所望のメール文書を検索する場合などには、効率が悪くなると考えられる。そこで、メールの読み上げの最中にメールツールに対するコマンドを発行可能にする。特に、そのコマンドは、音声入力によって行なえれば都合が良い。

【0234】まず、読み上げモードを設け、メールを読み上げる際の単位を設定可能にしておく。読み上げモードには、全文、段落、文の3つのモードがある。図63の右上の「読上」ボタンのとなりの「全文」の表示が読み上げモードを示す。「読上」ボタンにより、モードに従った音声合成を行なう。メール読み上げ時に使用する音声コマンドを、図68に示す。

【0235】ユーザは、モードを設定し、「読上」ボタンあるいは「読み上げ」と発声することにより、メールの読み上げを開始する。音声コマンド「ストップ」、「続行」により、読み上げの一時停止と再開を行なえる。「もう一度」は最後に読み上げた単位をもう一度読み上げる。「前の～」および「次の～」の「～」は読み上げの単位であり、メールツールはコマンドに従ってモードを自動的に変更する。例えばモードが「全文」の時に「次の文」と入力すれば、モードは自動的に「文」に変わる。「次」および「前」は、「次の～」および「前の～」の省略表現であり、それらのコマンドで扱われる単位はモードとして現在設定されている単位である。「速く」「ゆっくり」は読み上げ速度の設定、「高く」「低く」は読み上げ合成音の声の高さ設定、「男性」「女性」は合成音声の性別の設定を行なう音声コマンドである。

【0236】このように、メールの内容の音声による読み上げを可能にし、読み上げの制御を音声を使って行なうことで、マウスおよびキーボードのみを使って制御する時よりも、使い勝手が向上すると考えられる。特に、マルチウィンドウ環境において、聴覚と音声入力を音声

メールツールの制御に使い、視覚とキー入力を別のタスク（例えばテキストエディタ）に使うことで、1人のユーザによる複数のタスクの同時制御が可能となる。

【0237】音声合成機能は、メール文書の読み上げだけではなく、メールツールからユーザに対して提供されるメッセージにも利用可能である。例えば、マルチウィンドウ環境において、動作するメールツールがメッセージの出力に合成音声を利用する場合を考えてみる。まず、メールツールをその起動時にアイコン化しておく。メールツールが新規メールを受信すると、「××さんから新しいメールが届きました。未読分は全部で5通あります」といったメッセージを合成音声を使ってユーザに提供する。もちろんこのメッセージは、録音された音声データでも良いが、メッセージ文の変更し易さや、任意の数値データの読み上げを考えると、合成音声の方がメールツール等の応用プログラムの作成者にとっては都合が良い。新規メール受信通知のメッセージをいつも同じ様に出力するのではなく、例えば、メールに重要度を設定し、その重要度に従って音声メッセージを出力しなかったり、「××さんから緊急のメールが届きました」と、メッセージ文を変えたり、音声合成のパラメータを変更して声のトーンを変えることができる。メッセージとして、「サブジェクトは、会議通知です」と、メールの題についての情報を提供してもよい。このように、合成音声をメールツールのメッセージ出力に利用することで、ユーザは、メールツールを直接見ることなく、受信メールを読むか否かの決定ができる。

【0238】新規メール受信のメッセージは、ユーザが計算機上で行なっている作業に割り込むメッセージであり、ユーザの作業に割り込んで欲しいか否かは、作業内容によりけりである。例えば、何らかのプログラムのデモンストレーション中には、メールに割り込んで欲しくないであろう。そこで作業の重要度を設定し、作業の重要度とメールの重要度を比較して、メールの重要度が作業の重要度以上なら音声メッセージを出力し、それに満たない場合は出力しない、といった事を行なう。作業の重要度は、作業環境全体に設定したり、個々のプログラムに設定したり、プログラム内のサブタスク毎に設定する事が考えられる。

【0239】作業の重要度とメールの重要度を比較し、メールの受信の通知方法を決定するために、音声メールシステムを図69に示す構成とする。メールシステム691は、メッセージ入出力部6911の介在によって、音声入出力システム692やウィンドウシステム693と接続されている。音声入出力システム692やウィンドウシステム693からのメッセージは、メッセージの内容に従い、メッセージ入出力部6911によってふりわけられ、そのメッセージを処理すべき所において処理が行なわれる。

【0240】電子メール処理部6912は、外部の公衆

回線やLANを通じ、電子メール文書の送受信や、受信したメールに対する処理を行なう。タスク重要度管理テーブル6913は、音声入出力システムに接続したすべての応用プログラムの作業の重要度を音声入出力システムから受け取り、管理する。このタスクの重要度と、受信したメールの重要度から、受信したメールをユーザに対してどのように知らせるかの役割も、電子メール処理部6912が担う。

【0241】この機能を実現するために第14実施例で述べた音声入出力システムの持つ応用プログラム管理テーブルを拡張し、項目として、タスク優先度を新たに設定する。図70に拡張した応用プログラム管理テーブルを示す。ここでは、シェルツールのタスク優先度が「2」、DTPシステムのが「5」に設定されている。

【0242】さらに、この応用プログラム管理テーブルに値を設定したり、値を読み取るためのメッセージとして、図71に示すメッセージを新たに設ける。また、タスク優先度変更のたびにその通知をメールシステムが受け取れるようにするために、入力マスクとして、タスク優先度変更マスクを新たに設ける。

【0243】メールシステムは、入力マスクとして、タスク優先度変更マスクと、入力タスク変更マスクを設定することにより、音声入出力システムに接続されているすべての応用プログラムのタスク優先度と、音声フォーカスの有無を得、図72に示すようにその情報をタスク重要度管理テーブルに動的に反映することが可能である。電子メールの優先度は、例えば、「Preference:3」のようなヘッダ情報をメール文書に付加し、メール自体に重要度を設定することも可能であるし、発行者毎にメールの優先度を設定しても良い。メールシステムの電子メール処理部は、電子メールを受信するたびに図73に示す処理を行なう。

【0244】この場合、音声フォーカスが1つのタスクに当たっているか調べ（ステップ7801）、YESならば音声フォーカスのあるタスクの優先度を選択し、NOならば音声フォーカスの当たっているすべてのタスクの優先度の平均を選択する。例えば、その中で一番高い優先度を選択しても良い。そして、これらがメールの優先度より低いか調べ（ステップ7804）、YESならば音声を使って通知し（ステップ7805）、NOならば何も通知しない（ステップ7806）。この場合、アイコンの表示を変化させたり、動画像を用いたりといった様々な方法をユーザへのメールの受信通知に用いることができる。

【0245】応用プログラムとして、メールシステム以外に、シェルツールとDTPシステムが、音声入出力システムに接続している時の画面の表示例を図74に示す。図74(a)は、タスク重要度管理テーブルが図72の状態の時の画面表示例である。ここで、重要度3を持つメールを受信したとすると、図73に示した処理に

よればここで音声フォーカスの当たっているシェルツールの重要度が、メールの重要度より高い（小さい値ほど重要度が高いと見做す）ため、メールシステムはメールの受信をユーザに通知しない。これに対して、タスク重要度管理テーブルが図75の状態にある（対応する画面表示例は図74(b)）時に、先ほどと同様に、重要度3のメールを受信した際には、メールシステムは「新しいメールを受信しました」という音声出力を行ない、メールの受信をユーザに通知する。また、通知と同時にメールシステムは、自身に対して音声フォーカスを設定することによってユーザの作業に割り込み、ユーザにメールシステムを使わせることが可能である。

【0246】このように、新規受信の通知などに関するメッセージを、メールの重要度や作業の重要度に従って変化させることで、ユーザの作業を疎外しない柔軟なインターフェースをユーザに提供できることになる。

【0247】（第16実施例）第15実施例におけるメール文書の読み上げ機能は、受信したメールの一部あるいは全部をその文面に対して何の変更も加えず、合成音声を使ってそのまま読み上げるものであった。この方法は、メール文書が少なく、おしなべて小さい場合には問題は少ないが、メールが多く、大きくなるに従い、その機能だけでは不十分である。

【0248】図77は、音声メールシステムの概略構成を示すもので、音声入出力システム821に接続される音声メールシステム822を電子メール処理部8221、文書要約部8222、メッセージ入出力部8223より構成している。この場合、図78に示すように文書要約部8222を音声メールシステム822の外に設けるようにしてもよい。

【0249】ここで、メールシステム822は、音声入出力システム821と接続してその音声入出力機能を用いる。電子メール処理部8221は、外部の公衆回線やLANを通じ、電子メール文書の送受信や、受信したメールに対する処理を行なう。文書要約部8222は、電子メールなどの文書を要約するシステムである。テキスト文を要約する技術としては、「石橋ほか、英文要約システム「DIET」、情報処理学会第48回全国大会、6D-9(1989)」や、「喜多、説明文を要約するシステム、情報処理学会自然言語処理研究会、63-3(1987)」などが知られており、この技術を応用して、文書要約部を構成できる。

【0250】文書要約部8222は、電子メール処理部8221から要約前のメール文書を受け取り、要約して返す。電子メール処理部8221は、受信したメールの重要度や、文書の長さや文書の内容などに従って、そのメール文書を要約するか否か、また、どのような要約を行なうかを決定し、要約方法の情報とともにメールを文書要約部にひきわたす。電子メール処理部8221は、メールを受信するたびに、例えば図79に示すような処

理を行ない、受信メールに対する要約方法を決定する。

【0251】この場合、メールの重要度が「3」以上か判断し(ステップ8401)、「3」以上であれば要約しない(ステップ8402)。「3」以上出なければ、メール本分中に「至急」を含むか調べ(ステップ8403)、「至急」を含めば、さらに文書が長いかわ(ステップ8404)、文書が長くなければ要約せず(ステップ8402)、文書が長ければ要約する(ステップ8405)。また、本分中に「至急」を含まなければ、先頭行だけを要約する(ステップ8406)。そして、メー

ールに従った要約処理を行う(ステップ8407)。
【0252】メールのような文書の場合、その内容が完結していなかったり、短すぎたりして、要約に適さないこともあると考えられるが、その場合には、短いメールに対しては、要約を行なわない(必要がない)こともできるし、完結していなくて要約に失敗したメール文書に対しては、例えば、最初や最後の数行を取り出して読み上げるようにすれば、すべてのメールに対して何らかの要約処理をほどこすことができるといえる。要約は、例えば、音声による「要約」コマンドの形でユーザが指示することによってもできるし、あるいは、メールシステムが、受信メールの全てを(あるいは長いものだけを)自動的に要約しておくことによってもできる。

【0253】このように、音声メールツールがメール文書の要約機能をもつことで、メール文書処理の効率化をはかることができ、時に多忙なユーザや、大量のメールを処理する必要のあるユーザにとっては、便利である。

【0254】(第17実施例)第15および第16実施例においては、音声入出力システムの提供する音声認識および合成機能の利用に関して、音声メールツールを使って述べた。

【0255】これらは、GUIおよび音声出力を使ってユーザに情報を提供していたが、電話インターフェースなどの、GUIを利用できない環境において第15および16実施例で述べた機能はより有用である。本実施例では、GUIを利用しない電話を介した音声入出力インターフェースについて、音声メールシステムの例を使って述べる。

【0256】図80は第17実施例の概略構成を示している。この場合、音声認識システム851に接続される音声メールシステム852にメールアドレステーブル853を接続している。

【0257】この場合、音声入出力システム851は電話回線に接続されるが、この電話回線との接続は、既存技術を使えば可能であり、ここでは述べない。電話からの音声メールシステム852への入力、音声およびプッシュボタンにより行なえるとする。

【0258】メールは個人情報であるため、電話でメールの内容を確認する前にあらかじめ個人情報の認証手続

が必要である。これは電話のプッシュボタン等で行なうかパスワードの音声認識、あるいは話者照合技術により行なう。

【0259】認証手続において利用者を確認した後、音声認識を利用して、メールへのアクセスを対話的に進める。ここで述べる音声メールシステム852は、第15、16実施例で述べた音声認識と音声合成の機能が全て利用できる。即ち、音声入力によって、音声メールの全部のあるいは部分のあるいは要約された内容の確認を行なうことができる。音声メールシステム852の操作は基本的に全て音声を使って行なう。従って、メール送信も音声により行なう。電話インターフェースでは、プッシュボタンを使ってメールの内容を入力するのは現実的ではないため、メールの内容自体も音声となる。音声によるメール文書の作成は、音声認識と音声の録音を同時に行なうことで可能となる。図80の構成において、認識と録音を同時に行なうことは疎外されない。図81に、音声を使ったメール文書作成の例を示す。場面設定は、受信したメールの内容を音声(合声/肉声)により確認した後、そのメールに対して返事を出す所である。

【0260】まず、(1)のユーザの「録音開始」という音声を認識し、メールシステムは続くユーザの音声(2)の「〇〇です〜お伝え下さい」をメール文書として録音する。(2)の最後の「ストップ、ストップ」は、録音を停止する命令である。「ストップ」が繰り返されているのは、メール本文中の「ストップ」と、命令としての「ストップ」を区別するためである。「ストップ、ストップ」全体を認識対象語彙としても良い。メールシステムは「ストップ、ストップ」の区間を録音されたデータからカットする。ユーザは(3)の「内容確認」によりメール文書の内容(4)を確認し、(5)の「送信」によりメールを送信する。最後に(6)のメッセージにより、メールの送信を認識する。

【0261】ここで、(2)でユーザがデータを録音する際に、音声認識システムの音声認識部の中の音声検出部に音声データの先頭を検出させれば、「録音開始」から本文の入力までの間に間があいても、その無音区間を録音せずに済む。

【0262】また、録音終了を指定するための「ストップ、ストップ」などの単語の代わりに「送信」と発声し、「送信」を認識したならば、録音内容をメールデータとして自動的に送信してしまうこともできる。こうすれば、録音の停止を指定する「ストップ」の発声が不要になり、簡単にメールを送信できる。この時、送信したメールの内容は、「内容確認」などの発声により確認しなくとも、自動的に録音内容を再生することによって確認できる。

【0263】また、「録音開始」後、1つの音声区間を録音するようにすると、「ストップ、ストップ」のような録音停止命令は不要となる。音声区間の終端は、例え

ば「3秒間無音であれば音声データの入力終了とみなす」のように、余裕をもたせた設定にすれば、ユーザが一息でメッセージを入力しなければならないというような制約が緩和される。

【0264】このように、データとしての音声区間を検出するために、応用プログラムと音声認識システムとの間のメッセージとして、図82のメッセージを追加する。この音声区間検出メッセージは往復メッセージであり、図83に示すような手順でもって、音声区間のデータを入力音声から切り出すことができる。音声区間検出メッセージでは、パラメータとして、音声の終端を検出するための時間（例えば、3秒間無音区間が続いたら、その無音区間の前を音声区間とみなす）や、入力音声がない場合のタイムアウト指定（要求を発信してから30秒たったら、音声区間は検出されなかったとみなす）ができる。

【0265】また、メール文書の題は、ここで述べたように、受信メールに対して返事を出す際には、UNIXメールでの表現でいえば、受信したメールの“Subject:hello”に対しては“Subject:re:hello”のように、返事としての題を入れられるといえるが、電話口で新規にメールを作成する場合には、そのメールに題を付けられない。それを可能にするために、音声認識を組み合わせる。例を図84に示す。

【0266】この場合、ユーザの(1)「サブジェクト」という音声をメールシステムが認識すると、メールシステムは、サブジェクト入力モードになる。このモードでは、予め決められたサブジェクト(題)用の単語が認識対象語彙となる。例えば、「こんにちは」「お知らせ」「至急連絡下さい」「ごくろうさま」「会議通知」などが考えられる。図84の例では、(2)「会議通知」を入力する。メールシステムは「会議通知」を認識すると、メール文書にテキスト“Subject:会議通知”を挿入し(3)、(4)のような確認のメッセージを合成音声により行なう。

【0267】サブジェクト入力モード時の認識結果をうけて行なうのは、メールの題の挿入だけではなく、例えば、定型的なメール文書の入力が可能である。図85は、「ごくろうさま」という入力に対して、メールの本文として挿入される定型メールの例である。文書中の{receiver}と{sender}は、受信者、発信者の代入される変数を表している。この変数により、誰にでも同一の文面のメールを音声だけで送信できる。定型メールをデータベース化し、そのデータを音声で呼び出すことが可能であれば、便利であると考えられる。

【0268】また、第15実施例においては、メール文書中の任意の場所に音声データを追加・挿入可能としたが、サブジェクト入力モードにおいて、サブジェクト自体に音声データを付けることを可能とし、例えば、メールの受信と同時に音声サブジェクトを出力すれば、メー

ルの発信者やメールの内容が受信者に伝わり易いと考えられる。もちろん、音声認識によるサブジェクトの挿入と音声サブジェクトの録音を同時に行なっても良い。

【0269】受信メールに対する返事を送信するのではなく、電話口から送信先を指定するのには、音声認識を利用する。そのために、あらかじめ、学習機能を応用して単語登録を行ない、認識対語語彙とメールアドレスを結びつけておく。例えば図86に示すような外観を持つアドレスブックをメールシステムに持たせ、図87に示すメールアドレスの登録機能によって、メールアドレスと音声を結びつける。この時の登録の手順は、

遙メールアドレスブック(図86)を開く
遙登録用ウィンドウ(図87)を開きメールアドレスの新規登録を開始する。

遙名前とアドレスをキーボードで入力する。

遙学習に必要な回数(数回~数十回)、新しい単語(この例では鈴木)を発声する。

遙OKボタンを押し、登録を完了する。

【0270】このようにして、認識対語語彙(鈴木)と、メールアドレス(Suzuki@aaa,bbb,ccc,co.jp)とを結びつけておき、電話口で利用する。例えば、図88の手順で行なう。まず(1)でユーザが「送信先」と発声し、認識されると、メールシステムは、(2)のメッセージを音声出力し、ユーザに確認をとる。(3)では図A、Bなどによって登録された語彙が認識対象となっており、この例では、「鈴木」が認識されると、メール文書中にto:Suzuki@aaa,bbb,ccc,co.jpが挿入される。

【0271】(4)(5)はメールアドレスの認識の様子を示している。(4)の「鈴木」の音声のように、例えば図87における登録の際に利用した音声の1つを自動的に録音しておき、認識の確認に使うことができる。

【0272】(4)の「Suzuki@…」は、合成音声によるアルファベット読み下しを使って確認を行なう例である。

【0273】この方法では、音声によるメールアドレスの指定は、予め登録したものにしか適用できないが、次に述べるように、予め登録しているメールアドレスを音声を使って指定することができる。そのためにまず、ユーザが過去に受け取ったメールから、自動的にメールアドレスのデータベースを作成する機能を付ける。メールアドレスは、UNIXメールにおいては、メールのヘッダとして入っており、そこからデータベースを作成するのは困難ではない。メールアドレスの構成は、例えば、ユーザ名@部課名、組織名、組織区分、国の区分のような構成になっており、メールアドレスの逆順(国→ユーザ名)にツリー状の階層構造を持つデータベースが作成できる。

【0274】メールシステムは、国の区分から順に、図89のように合成音声による読み上げを用いて、メール

アドレスを順にたどっていく。図 89 の例において、あやまったノード（メールアドレスを順にたどった際の節）を選択した際は「取り消し」などの語彙をもって、1つ前の（上位の）ノードに戻ったり、「取り止め」などの語彙をもって、アドレスの入力を取り止めたりできる。また、任意のノードに認識対語語彙を予め結びつけておき、例えば、会社名を発声することで、その会社のメールアドレスノードまで、一時に移動することもできる。

【0275】このような方法をとれば、過去にメールをくれた人に対してならば、音声をつかってそのメールアドレスを指定することが可能となる。

【0276】また、単語単位の認識辞書が不要な音韻認識をベースとした音声認識システムが広く研究されているが、これを用いることにより過去に届いたメール中に該当するアドレスがない場合でも、音声によってアドレスを入力し、メールを転送することが可能である。

【0277】（第 18 実施例）本発明の第 1 実施例や第 14 実施例で述べた音声認識インターフェースでは、音声認識システムあるいは音声入出力システム専用開発した応用プログラムを対象として、音声認識や音声合成のサービスを提供するものであった。本実施例では、上記のような専用プログラムに対する音声による制御に加えて、前記音声認識システムあるいは音声入出力システムと直接メッセージをやり取りすることのできないような任意の応用プログラムに対する音声による制御を可能とする拡張を前記音声認識インターフェースに施すものである。これによって、音声認識の応用分野とユーザの拡大を図ることができる。本実施例では、第 14 実施例に上記拡張を施した例を説明するが、同様の拡張を第 1 実施例に施すことが可能であることは明らかである。

【0278】以下、本実施例について説明する。図 90 は、本実施例の音声入出力インターフェースの全体構成であり、第 14 実施例で述べたものと同一の音声入出力システム 1 と、そのメッセージ処理部 11（図示せず）に應用プログラムとして接続された音声インターフェース管理システム（以下、SIM と呼ぶ）104 からなる。

【0279】汎用應用プログラム（以下、GAP と呼ぶ）103 は、音声入出力システム 1 と直接接続されていない應用プログラムであり、音声入出力システム 1 とは全く独立して動作可能なプログラムである。これに対して、専用應用プログラム（以下、SAP と呼ぶ）102 は、音声入出力システム 1 と直接接続して動作するものである。

【0280】SIM104 は、SAP の一つであり、音声入出力システム 1 と GAP103 との仲立ちをして、GAP103 に対する音声による操作を可能にする應用プログラムである。音声フォーカスの表示も、SIM104 が行なう。なお、SAP102 は、図 50 の應用プ

ログラム 2 に対応するものである。SAP および GAP は、1つの音声入出力システムに対してそれぞれ複数個存在することが可能である。

【0281】次に、SIM104 による、GAP103 に対する操作について説明する。GAP103 は、SAP102 と異なり音声入出力システムと直接接続されてはならず、GAP103 が受け付けられる入力、音声以外のキーボードやマウスといった入力装置からのものである。従って、SIM104 は、音声による GAP103 の操作を実現するために、音声入力を GAP103 の受理できる形の入力、例えばキーボード入力やマウス入力等に変換する。本実施例では、SIM104 は、図 90 に示すように、音声インターフェース管理部 141、プログラム操作登録部 142、メッセージ変換部 143 から構成される。音声インターフェース管理部 141 内には、應用プログラムごとの音声認識結果と操作との対応表が設けられており、この対応表（以下、音声インターフェース管理テーブルと言う）の情報は、プログラム操作登録部 142 によって登録される。前記メッセージ処理部 11 と直接接続されるメッセージ変換部 143 は、音声入出力システム 1 とのメッセージのやり取りを行なう機能、つまり図 6 のメッセージ入出力部 21 の機能を包含するものであり、認識結果を受信した際に、音声インターフェース管理テーブルを参照して、該認識結果を GAP103 に対する操作コマンドに変換し、GAP103 に送信する。

【0282】SIM104 から GAP103 に操作コマンドを送るには、GAP103 自身が他のアプリケーションからの操作の手段を提供していなければならない。

【0283】ウインドウシステムを利用したアプリケーションであれば、SIM104 はウインドウシステムを介して、その GAP103 へキーやマウスなどの入力デバイスによる操作コマンドの入力時に発生するのと同じメッセージを GAP103 に送る。このようなメッセージ送信の方法は X ウインドウシステムなどの各ウインドウシステムの提供するライブラリにある機能で容易に実装できる。実際、ウインドウシステムでは、メッセージの送付先が GAP103 そのものではなく、GAP103 の中で生成したウインドウなどのオブジェクトの場合もある。メッセージ送信時に、そのオブジェクトの識別子である必要があるケースもあるが、後述するプログラム操作登録の内容や、ウインドウシステムに問合せて識別子の情報から、送り先のオブジェクトの識別子を決定することは容易である。

【0284】次に、具体例をあげて説明する。図 91 に示すように、1つの音声入出力システム 1 に対して、音声インターフェース管理システム 104 とメールツール 120 が直接接続して動作し、また音声入出力システム 1 と直接接続できない GAP であるシェルツール 130 とエディタ 131 が並行して動作しているとする。この

ときの画面表示は、例えば図92のように行なえる。

【0285】この場合のSIM104の音声インターフェース管理テーブルの一例を図93に示す。この表における“プログラム名”は、認識対象語彙であり、ユーザがプログラム名を発声することで応用プログラムに対する疑似音声フォーカスを切り換えることができる。“応用プログラム”は、応用プログラム自体の識別子であると共に、コマンドの送信対象を表す。

【0286】上記の疑似音声フォーカスは、応用プログラムに対して疑似的に設けた音声フォーカスである。GAPは音声入出力システム1と直接接続しておらず、従って、音声入出力システム1はGAPの存在を関知しないため、GAPに対して本当の音声フォーカスは設定されない。SIM104は、「シェルツール」や「エディタ」等、GAPの名前を認識結果として受け取ると、そのプログラムについて定義されているコマンド名を認識対象語彙とする設定要求を、音声入出力システム1に対して行なう（例えば、「シェルツール」の場合、「エルエス」や「プロセス」）。そして、図12や図19等で示したような音声フォーカスの表示をそのプログラムに対して行なう。

【0287】図94に示すように、GAP103に関係する真の音声フォーカスはSIM104に設定され、実際に画面に表示されるのは疑似音声フォーカスである。SIM104が、プログラム名の認識をきっかけにして、認識のコンテキストを切り換えるのである。なお、メールツールにみるように、SAPの疑似音声フォーカスと真の音声フォーカスは合致する。

【0288】SIMおよびGAPのコマンド名の属性は、SIMに対してローカルである。すなわち、SIMに音声フォーカスが設定されているときに認識対象となる。SAPにコマンドを送信する際、SIM104に音声フォーカスが設定されない状態であるため、SAP102に関するコマンド名は、グローバル属性を持つ。例えば、図93のメールツールのコマンド名「終了」の属性がグローバルである。なお、図93で、ローカル、グローバルといった認識対象語彙の属性は、プログラム名および認識対象語彙の欄の括弧内に示されている。属性値は、“0”がローカル、“1”がグローバルである。

【0289】このようなメッセージ変換部143の処理手順の一例を図95に示す。すなわち、音声入出力システム1のメッセージ処理部11から受信した認識結果がプログラム名である場合、直前の疑似フォーカスに関するコマンド名を認識対象からはずし（ステップ9003）、認識したプログラム名を持つ応用プログラムに疑似フォーカスを設定し（ステップ9004）、その応用プログラムのコマンド名を認識対象として設定（追加）する（ステップ9005）。

【0290】一方、受信した認識結果がプログラム名でない場合（ステップ9002）、コマンド名に対応する

コマンドを、疑似フォーカスの設定されている応用プログラムに送信する（ステップ9006）。

【0291】以上述べたように、本実施例のような構成をとることにより、既に存在する音声入力（認識）を用いない応用プログラム（GAP）に対しても、音声認識の利用が可能となり、ユーザの拡大と使い勝手の向上が実現できる。

【0292】（第19実施例）ウィンドウベースのGUI（グラフィカル・ユーザ・インターフェース）を持つシステム下では、1つのプログラムを複数のウィンドウを使って構成することができる。本実施例では、上記第18実施例をもとに、複数のウィンドウを持つ応用プログラムの個々のウィンドウに対する音声入力を可能にするべく、システムを拡張した例を説明する。これにより、よりきめ細かい音声認識の利用が可能となり、操作性が向上する。

【0293】これまで説明してきた実施例においては、音声入出力システム1によって音声フォーカスが設定可能な単位は、“応用プログラム”であったが、本実施例では、その単位を“音声ウィンドウ”とする。音声ウィンドウは、応用プログラム中に複数個作成可能であり、個々の音声ウィンドウは、音声ウィンドウ名、入力マスク、および認識対象語彙セットを持つ。

【0294】図96が、実施例14（図50参照）で説明した音声入出力システム1を拡張して、音声ウィンドウを扱えるようにしたものである。ここで、図96の応用プログラム管理テーブル13は、後述するように拡張する。また、応用プログラム2に音声ウィンドウ23が加わっているが、音声ウィンドウ23の実体は、音声入出力システム1の応用プログラム管理テーブル13中に存在する。

【0295】以下、具体例をあげて説明する。第18実施例と同様に、応用プログラムとして、SIM（104）、シェルツール、エディタ、およびメールツールの4つが動作しているとする。このうち、SIMとメールツールはSAPであり、シェルツールとエディタはGAPである。図97のように、シェルツールとエディタをそれぞれ2つのウィンドウから構成し、それ以外を1つのウィンドウから構成したとする。この場合の音声入出力インターフェース全体の構成を図98に示す。専用プログラム（SAP）であるメールツール120は、自分用の音声ウィンドウ223を持ち、SIM104は、自分用の音声ウィンドウ0（144₀）に加えて、汎用プログラム用の音声ウィンドウ1～4（144₁～144₄）を持つ。この音声ウィンドウは、図97に示すようないわゆるウィンドウシステム（図示せず）やOS（図示せず）におけるウィンドウとは異なり、ビジュアルな属性を持たないものである。ウィンドウシステムのウィンドウは、通常、ツリー構造を持ち、その構造やウィンドウシステムの内部状態の変化を応用プログラム内部か

ら知ることができる。SIM104は、そのようなウィンドウシステムの情報と、音声入出力システム1の情報にアクセスし、ウィンドウと音声ウィンドウとを結びつけて協調的に動作させ、統一的なユーザインターフェースを提供する。ウィンドウと音声ウィンドウとの結び付けは、両者にウィンドウ名などの一意かつ同一の属性を付与したり、プログラム操作登録部142で対話的に行なうことで可能である。

【0296】音声ウィンドウはその属性として、ウィンドウ名、認識対象語彙、入力マスク等を持ち、音声入出力システム1はこの音声ウィンドウ単位で音声フォーカスの設定を行なう。ウィンドウ名やコマンド名などの認識対象語彙の属性として、ローカル、グローバルに加え、ウィンドウを設ける。ローカル属性を持つ語彙は、それが属する音声ウィンドウに音声フォーカスが設定されている時に認識対象となる。グローバル属性を持つ語彙は、音声フォーカスがどこに設定されているように常に認識対象となる。ウィンドウ属性を持つ語彙は、それが属する音声ウィンドウに音声フォーカスが設定されていなくとも、その音声ウィンドウと同じ応用プログラムに属する音声ウィンドウに音声フォーカスが設定されている時に認識対象となる。

【0297】また、複数の音声ウィンドウをグループ化して認識語彙を混合し、認識結果に応じて自動的にその認識語彙の属する音声ウィンドウへ結果を送信することもできる。例えば、応用プログラム管理テーブルが図102の状態の場合に、シェルツールとエディタをグループ化してエルエス、プロセス、カット、コピー、ペーストを1度に認識し、エルエスまたはプロセスが認識された場合はシェルツールへ認識結果を送り、カット、コピー、またはペーストが認識された場合にはエディタへ認識結果を送るようにする。

【0298】これにより、シェルツールとエディタの間の音声フォーカスの移動を省略して効率的に両者の作業を行うことができる。複数の音声ウィンドウの語彙の中にも同じものがある場合には、それを語彙として持つ複数の音声ウィンドウへ同時に認識結果を送信しても良いし、音声フォーカスの当たっている音声ウィンドウを優先させることにしても良い。なお、グループ化は、図102の応用プログラム管理テーブルのグループ化IDの属性により、行うかどうかを決めることができる。

【0299】また、音声ウィンドウのグループ化の一方方法として、音声ウィンドウに親子関係を導入し、親ウィンドウと子ウィンドウをグループ化して両者の語彙を同時に認識することもできる。例えば、応用プログラム管理テーブルが図102の状態の場合に、シェルツールの設定ウィンドウに関して、その親のシェルツールウィンドウと設定ウィンドウをグループ化する。そして、設定ウィンドウに音声フォーカスが当たったときに両者の混合した語彙によって認識を行う。

【0300】これにより、子音声ウィンドウに音声フォーカスが当たっている場合に、音声フォーカスの移動を省略してその親ウィンドウへの音声入力を行うことができ、作業が効率化できる。なお、親ウィンドウと子ウィンドウで同じ語彙を持つ場合には、音声フォーカスの当たっている子ウィンドウに優先して認識結果を送るようである。

【0301】図98の状態の時、SIM104の音声インターフェース管理部141内の音声インターフェース管理テーブルは、図99のようになる。図93のテーブルにウィンドウIDを加え、プログラム名の替りにウィンドウ名を追加した形である。ウィンドウIDとは、ウィンドウシステムにおけるウィンドウの識別子である

(図97参照)。図99に示すように、ウィンドウIDと音声ウィンドウIDとは一対一に対応しており、この表を用いてSIM104はウィンドウと音声ウィンドウとを連動させる。例えば、この例でいうと「シェルツール」を認識したならば、SIM104はID=1の音声ウィンドウに音声フォーカスを設定し、ID=101のウィンドウの表示を図19に示したように音声フォーカスの設定された状態にする。

【0302】ウィンドウシステムやOSによっては、他の応用プログラムウィンドウの表示を変更できない場合があるが、そのときには図100の斜線部w1で示すような形で独立した別のウィンドウを他の応用プログラムのウィンドウに貼り付け、音声フォーカスの所在を示す。この外付けウィンドウの表示の例を図101に示す。図のように、応用プログラムの上部に音声フォーカスを示す表示(ウィンドウ)が示される。なお、このウィンドウの位置は、音声フォーカスが明示できればどこでも良く、また数もいくつでも良い。また、静止画だけでなく、動画像を使うことで、音声フォーカスの位置がより分かり易くなる。

【0303】ここで、図18で示した音声入出力システム1の応用プログラム管理テーブル13は、図102に示すように拡張される。新たな欄として音声ウィンドウIDおよびウィンドウ名が付加されている。音声ウィンドウIDは、音声フォーカスの設定されている音声ウィンドウの識別子であり、ウィンドウ名はその名前である。ローカル、グローバルといった認識対象語彙の属性は、ウィンドウ名および認識対象語彙の欄の括弧内に示されている。属性値は、“0”がローカル、“2”がグローバル、“1”がウィンドウである。音声入出力インターフェース1の構成が図98である場合の音声入出力システム1の応用プログラム管理テーブル13は図102に示す状態にあり、音声インターフェース管理システム104の音声インターフェース管理テーブルが図99に示す状態にある。この時、疑似音声フォーカスによって、ユーザには、音声フォーカスが“シェルツール”

(ウィンドウID=101)に設定されているように見

えている。一方、真の音声フォーカスは、ウィンドウ（ID=101）と対応付けられた所の音声ウィンドウ（ID=1）に設定されており、その音声ウィンドウは、SIM104に属している。例えば、この状態で認識可能な語彙は、「エルエス」、「プロセス」、「シェルツール」、「エディタ」、「メールツール」、「システム」、および「設定」である。

【0304】上記構成において、音声入出力システム1が認識処理を行い、その認識結果が、それぞれの語彙が設定されている音声ウィンドウに送られる。図103に、この認識処理の手順の一例を示す。

【0305】まず、ウィンドウ（0）について、音声フォーカスが設定されている場合、当該ウィンドウ（0）に設定されている語彙を認識語彙リストに追加する（ステップ9103）。一方、音声フォーカスが設定されていない場合、当該ウィンドウ（0）が音声フォーカスの設定されている音声ウィンドウと同じ応用プログラムに属するときは、当該ウィンドウ（0）の語彙のうち属性値が“1”であるものを認識語彙リストに追加し（ステップ9105）、属しないときは、当該ウィンドウ（0）の語彙のうち属性値が“2”であるものを認識語彙リストに追加する（ステップ9106）。

【0306】以上の処理を、ウィンドウ（1）をはじめとする他の全ウィンドウについて行う。

【0307】そして、認識処理を行い（ステップ9108）、第1位の認識結果がウィンドウ名である場合、第1位の語彙が設定されていたウィンドウに音声フォーカスを設定し（ステップ9110）、ウィンドウ名でない場合、第1位の語彙が設定されていたウィンドウに上記認識結果を送信する（ステップ9111）。

【0308】例えば、図102において、認識可能な語彙の1つである「設定」の設定されている音声ウィンドウは2つ（ID=2とID=4）あるが、それぞれの語彙の属性が“1”（＝ウィンドウ）であることから、ここで認識した結果「設定」は、音声ウィンドウID=2に送られる。これに対して、音声フォーカスが音声ウィンドウID=3に設定されている場合に認識された「設定」は、音声ウィンドウID=4に送られる。ウィンドウ名を認識した際に音声入出力システム1の動作としては、単に認識結果をウィンドウ名の属する音声ウィンドウに送ることもできるし、送らずに音声フォーカスをその音声ウィンドウに設定することもできる。

【0309】このように、認識対象語彙にウィンドウ属性を持たせることで、複数の応用プログラムのウィンドウに同一の名前を付け、操作することが可能となる。本実施例により音声認識インターフェースとしての使い勝手が大幅に向上する。

【0310】（第20実施例）第18実施例および第19実施例で述べたように、音声認識システムからの音声メッセージを音声インターフェース管理システム104

によって音声メッセージを変換して送信することにより、音声入出力インターフェースと直接通信する手段を持たない既存の応用プログラムに対しても、音声入力を行なうことが可能になった。

【0311】既存の応用プログラムに本発明の音声入出力インターフェースを適用する場合には、既存のプログラムの操作と、それを行うための語彙との対応を、音声入出力インターフェース専用の応用プログラムとは別個にとる必要がある。この実施例では、“語彙”と“プログラムの操作”との対応をとるためのプログラム操作の登録について説明する。

【0312】プログラム操作の登録では、音声フォーカスを目的の応用プログラムに移動させるのに用いるプログラム名またはウィンドウ名の登録と、既存の応用プログラムの操作を行なうためのキー入力またはマウス入力イベントの系列と語彙との対応づけを行なう。例えば、シェルツールのウィンドウを2つ使う場合には、ウィンドウ名として「シェル1」、「シェル2」と付け、シェルツールの中で行なう操作、例えば画面上の文字を全部消去するクリア（clear）コマンドを行なうためのキー入力系列に対し「クリア」という単語を割り当て、登録する。

【0313】通常、一般の応用プログラムは、そのプログラムが表示しているウィンドウのウィンドウ名を持っていないため、名前でウィンドウを指定するためには、ウィンドウに名前を付け、音声インターフェース管理テーブルからウィンドウ名で対象ウィンドウを識別できるようにすることが必要である。このため、第19実施例の図99に示すように、音声インターフェース管理テーブルに、ウィンドウシステムにおけるウィンドウ識別子であるウィンドウIDとウィンドウ名とを格納するフィールドを持たせている。このテーブルにより、音声インターフェース管理部141は、例えば「エディタ」が認識結果として送られて来た場合には、ウィンドウID103を持つウィンドウに対して疑似音声フォーカスを設定する。上述のウィンドウIDは、ウィンドウシステム（図示せず）の持つ情報にアクセスすることにより得られる。例えば、ウィンドウシステムのサーバー（図示せず）に、ウィンドウ構造に関する情報を問い合わせることにより得ることができるが、ウィンドウ名も同時に得られるとは限らない。ウィンドウIDとウィンドウ名を同時に得るにはウィンドウ名を指定してプログラムを起動する方法があるが、既に動作中のプログラムが別のウィンドウを新たに生成するポップアップウィンドウのような場合には、起動前から名前を付けることは困難である。そのような場合には、マウスでウィンドウをクリックすることによって、そのウィンドウのウィンドウIDを獲得し、そのウィンドウIDにウィンドウ名を対応させるという方法でウィンドウ名を付けることができる。マウスがクリックされたウィンドウのIDは、ウィンド

ウシステムのサーバーに問い合わせることで容易に得られる。

【0314】次に、ウィンドウへの名前付けとプログラム操作の登録方法について以下に説明する。図104は、前記プログラム操作登録部142の構成である。このプログラム操作登録部142は、登録内容の画面への表示とユーザからの入力を行なうプログラム操作表示編集部151と、登録内容をファイル200に保存する登録内容保存部152と、ウィンドウシステムからウィンドウIDを取得するウィンドウID取得部153からなる。

【0315】プログラム操作表示編集部151は、例えば図105のような登録画面を表示してウィンドウ名やプログラム操作、単語名などの入力を行ない、前記音声インターフェース管理部141内の音声インターフェース管理テーブルに登録内容を書き込む。登録内容保存部152は、プログラム操作の登録内容をファイル200に保存する。ウィンドウIDの取得は、ウィンドウシステムのサーバーに問い合わせることにより容易に行なえる。

【0316】図105の登録画面は、プログラム操作登録内容を音声インターフェース管理テーブルに書き込む「登録」ボタン、入力内容を取り消して入力前の状態に戻すための「取り消し」ボタン、登録を終了するための「終了」ボタン、対象とする一般応用プログラムのウィンドウIDを取得するための「ウィンドウID取得」ボタン、応用プログラムの種類を入力する「応用プログラムクラス」(APクラス)ウィンドウ、ウィンドウ名を入力する「ウィンドウ名」ウィンドウ、および語彙とそれに対応したプログラム操作を表すキー入力系列またはマウス入力系列を入力するプログラム操作入力ウィンドウからなる。

【0317】図105では、応用プログラムクラスとして「シェル」、シェルのウィンドウ名として「シェル1」が選択され、背景色が反転しており、シェル1に対する操作として単語「エルエス」と「クリア」に相当するキー入力操作と、それらの語彙のスコープとしてローカル(0)が、編集用ウィンドウに入力された状態を示している。

【0318】次に、プログラム操作の登録手順について図106を用いて説明する。プログラム操作登録部142は、メッセージ変換部143から起動され、まず、プログラム操作登録内容を保存した登録内容ファイル200から登録内容を読み出し(ステップ9201)、画面表示を行ないユーザの入力待ちの状態(ステップ9202)になる。

【0319】ここで、ユーザが、APクラス、ウィンドウ名、語彙、プログラム操作などの入力、あるいは、登録ボタン、取り消しボタン、終了ボタン、ウィンドウID取得ボタン等の入力を行なう。

【0320】入力が登録ボタンであった場合には(ステップ9203)、画面に表示されている編集結果を保存ファイル200へ保存し、更に音声インターフェース管理テーブル141へ書き込んで登録内容を音声入出力インターフェースの動作に反映させる(ステップ9204)。

【0321】入力を取り消しボタンであった場合には(ステップ9205)、再度、保存ファイル200から登録内容を読み込んで表示し、入力待ちの状態に戻る(ステップ9202)。

【0322】入力が既に登録済みの応用プログラムクラス(APクラス)であった場合(ステップ9206)選択されたAPクラスのウィンドウ名の一覧と語彙、プログラム操作を画面表示し(ステップ9207)、入力待ちの状態に戻る(ステップ9202)。

【0323】入力がウィンドウID取得ボタンであった場合(ステップ9208)、まず、ウィンドウ名が選択されているか判別し(ステップ9209)、選択されていない場合には入力待ちに戻り(ステップ9202)、選択されている場合にはマウスでウィンドウがクリックされるのを待ち、クリックされたウィンドウのIDを取得して、図99に示すような音声インターフェース管理テーブルに選択されているウィンドウ名とウィンドウIDを書き込む(ステップ9210)。

【0324】入力が終了ボタンである場合(ステップ9211)には、画面表示内容の内容の音声インターフェース管理テーブルへの書き込みとファイル200への保存を行なって(ステップ9212)、登録を終了する。

【0325】以上述べたように、プログラム操作登録の際、応用プログラムの種類を指定することにより、同一のプログラム操作を入力せずに、自動的に指定することが可能になり、登録が効率的に行なえるようになる。

【0326】また、名前を指定して起動することが困難な応用プログラムのウィンドウに対しても、マウスのクリックされたウィンドウのIDを取得してウィンドウ名と結び付けるようにすることにより、容易にウィンドウ名を付けて音声入力を行なえるようになる。

【0327】上述の登録の例では、すでに生成されているウィンドウのIDを利用して、操作コマンドと認識結果の対応をとっていたが、一般にウィンドウ等のオブジェクトIDは生成時に決定され、同じ種類のアプリケーションであっても異なるIDが付与される。したがって、登録時にウィンドウ階層やウィンドウ名など、同じ種類のアプリケーションで共通のウィンドウ属性値をウィンドウシステムに問い合わせて登録内容に付加しておけば、これらの属性値を照合することによって同種のアプリケーションで共通に登録内容を反映させることができる。

【0328】さらに、この登録時に、登録対象のアプリケーションに複数のウィンドウ名を登録しておくことに

よって、同じ種類のアプリケーションが起動された時に（既に使われている音声ウインドウ名を音声認識システムに問合わせて）使われていないウインドウ名を起動されたアプリケーションの音声ウインドウ名として利用すれば音声ウインドウ名の衝突を避けることができる。

【0329】（第21実施例）次に、音声入出力インターフェースにおいて音声の認識を行なうための認識辞書の編集機能に関する実施例について説明する。

【0330】図107は、辞書編集部144を持つ音声インターフェース管理システム104の構成である。辞書編集部144は、メッセージ変換部143から起動され、編集を終了すると終了メッセージをメッセージ変換部143へ返す。この終了メッセージを受けて音声インターフェース管理部141は、音声入出力システム1へ、編集した後の新しい辞書のロード命令を出すことができる。

【0331】ここで、図108は、認識辞書の構成の例である。認識辞書には単語ごとに、パターンマッチング用のテンプレートの他、単語名や単語ID、あるいは認識パラメータ等のデータがヘッダに格納されている。これらのデータの内容を表示し、編集する機能を備えることにより、使わない単語の辞書を削除して辞書に要する実行時のメモリ量を減らしたり、単語名やIDを付け替えたりすることが容易に行なえるようになる。

【0332】次に、辞書編集部144の構成について説明する。辞書編集部144は、図109に示すように、辞書内容を表示してユーザが編集を行なえるようにする辞書内容表示編集部441と、辞書内容のチェックや検索を行なう辞書内容検索部442からなる。

【0333】辞書内容は、例えば図110のような画面に表示される。画面中には、辞書名を表示する辞書名ウインドウ、語彙番号、単語ID、単語、パラメータ、辞書番号を表示する辞書内容ウインドウ、辞書の削除を行なう「削除」ボタン、パラメータの検索を行なう「検索」ボタン、内容の全表示を行なう「全表示」ボタン、辞書編集を終了する「終了」ボタン、辞書内容チェック結果を表示するステータスウインドウ、検索の際の値を入力する検索値ウインドウなどがある。辞書内容ウインドウのパラメータの項目はメニューになっており、マウスでクリックすると図に示すようなパラメータ内容が表示されて表示する内容を選択するようになれる。

【0334】辞書内容のチェックは、辞書名を選択したときに自動的に動作するようでき、例えば、同じIDの単語がないか、あるいは同じ単語名の辞書がないか等のチェックや、認識パラメータのくい違いがないか等のチェックが行なわれ、結果がステータスウインドウに表示される。

【0335】図110の項目では、辞書として、“common”および“usr.1”というファイル名の辞書が選択され、辞書内容としてその2つの内容がマージして表示さ

れる。例えば、語彙No. “1”はID=1のオープンで辞書作成に使ったデータ数が100であることを示している。また、語彙No. “2”はID=2のクリアでこの単語が選択されて背景色が暗く変わっていることを示している。

【0336】次に、辞書編集の処理の手順を、図111を用いて説明する。辞書編集部が起動されるとまず、辞書ファイルから辞書内容を読み出し（ステップ9301）、画面に内容を表示して入力待ちする状態になる（ステップ9302）。

【0337】入力が削除ボタンであった場合には（ステップ9303）、ユーザが指定した辞書No.の辞書をファイルから削除し（ステップ9304）、入力待ちに戻る（ステップ9302）。

【0338】入力が全表示ボタンであった場合には（ステップ9305）、辞書内容を再度読み出して（ステップ9301）、入力待ちに戻る（ステップ9302）。

【0339】入力が検索ボタンであった場合には、パラメータメニューからのパラメータの指定を待ち（ステップ9307）、指定されたパラメータと検索値ウインドウに入力された値に合致する辞書のみ辞書内容として表示して（ステップ9308）、入力待ちに戻る（ステップ9302）。

【0340】入力が終了ボタンであった場合には、画面に入力した内容から辞書ファイルを更新し（ステップ9310）終了したことをメッセージ変換部へ知らせ（ステップ9311）終了する。

【0341】以上に述べた辞書編集部により、不要な単語辞書の削除や内容の確認、単語名の変更などの編集が容易に行なえ、また同じIDや単語の2重使用や認識パラメータの不統一のチェック等が容易に行なえる。

【0342】（第22実施例）本発明の第18、19実施例で述べた音声入出力インターフェースでは、ユーザの発声の認識結果の確認および認識結果により引き起こされる応用プログラムの動作の確認は、応用プログラムの提示する画面情報を通じて行っている。例えば、認識結果（および認識失敗）を文字情報としてユーザに提示する。「シェルツール」などプログラム名を呼んだ時にシェルツールの表示を第19実施例の図100、101のように変更する。「アイコン化」の発声に対して、音声フォーカスの当たったウインドウをアイコン化する等、音声による応用プログラムへの働きかけは、応用プログラムの行う画面表示の変化としてユーザへフィードバックされる。しかし、応用プログラムによっては、操作によりその表示が殆んどあるいは全く変化しない事も考えられる。また、キーボードフォーカスと音声フォーカスを分離できるという本発明の特長を生かして音声フォーカスを当てた応用プログラムを表示しない状態で使用することも考えられる。このような場合には、認識結果やそれによる操作の確認を画面出力ではなく、第14実施

例で述べた、音声合成機能を利用した音声出力によって行うことで、ユーザの応用プログラム操作上の利便性が向上する。

【0343】動作確認を音声出力によって行うために、第19実施例の音声インタフェースマネージャ（図98）を図112のように拡張する。すなわち、音声インタフェース管理システム（SIM）に応答音声管理部401と応答音声登録部403を追加する。

【0344】ユーザの行った発声に対してどのような応答音声を出すかを定義するのが、応答音声管理部401であり、その登録を行うのが応答音声登録部403である。そして、動作（すなわちメッセージ）が発生した際に応答音声管理部401を参照して音声応答を出力するのが、メッセージ変換部143である。

【0345】応答音声管理部401の例を図113に示す。応答音声管理部401は、音声応答を出力するきっかけとなる動作と、動作時に行う応答コマンドおよび、その設定を実際に適用するか否かを決定するフラグから成る。動作は、音声によらないものでもよい。応答には、コマンドが記述される。synth（）は、その引数をテキストとして合成音声出力するコマンド、play（）は、引数を波形データと見做し、出力するコマンドである。

【0346】メッセージ変換部143は応答音声管理部401のデータを参照し、図114に示す流れにより処理を行う。まず、音声入出力システムから受信したメッセージが認識結果が否かを判定し（ステップ10001）、認識処理が成功したか否かを判定する（ステップ10002）。ついでその成功・失敗に応じて、音声応答コマンドを実行する（ステップ10003、ステップ10004）。ステップ10005は、認識処理の成功・失敗以外の応答音声出力する段階であり、図113の3行目以下の設定にあたる。この流れに従えば、認識はできたが類似度が低い、あるいは音声入力レベルが大き（小）すぎるなどの理由によって、認識失敗した際には、「えっ？」などという音声データが出力され、応用プログラム名、例えば「メール」が認識された時には、合成音声により「はい、メールです」などと出力される。ここで、図113中の\$<cat>は、認識結果の語彙名が置換される）。

【0347】応答音声管理部401のコマンドを登録するのが、図115に示す応答音声登録部403である。各動作に対してコマンドを記述し、また適用するか否かのチェックボックスをチェックし、OKボタンを押すことで登録を確認する。

【0348】応答音声管理部403の応答コマンドは、メッセージ変換部143が処理するものであり、第19実施例の図99に示した音声インタフェース管理テーブルのコマンドとして記述できる。ここにplay（）およびsynth（）コマンドを記述することで、音声入

出力システム1と直接情報を交換できないGAPの動作に対して、その応用プログラムに即した応答音声出力を定義できる。

【0349】このように、音声入力によって行われる（あるいは行われない）動作に対して、動作毎に意味のある音声応答を返す機構をSIMに設け、音声入力に対しては音声で応答するという自然な方法で、ユーザが画面の表示の変化を注視しなくとも（あるいは全くみなくとも）応用プログラムの実行した動作を確認できるため、音声入出力インタフェースの操作性が向上する。

【0350】（第23実施例）本発明の第9実施例では、認識辞書作成のためのデータ収集について説明したが、収集データの中には、間違った語彙の発声や音声区間の検出誤りなどにより、誤りデータが含まれることがある。例えば「ひらく」という単語は「く」の音が小さく発声されることがあり、「く」が抜けて「ひら」のみ音声区間として検出されることがある。このような誤ったデータによる認識辞書の学習は認識精度を大きく低下させるため、データの確認を行って誤りデータを取り除くことが必要である。そこで本実施例では、データの確認を容易に且つ確実にこなえるように、音を再生して聞くことによりデータ確認するようにしている。

【0351】従来、収集した音声データを再生して確認する方法では、検出された音声区間のみを再生する場合が多いが、語彙によっては、音声の始末端が誤って検出されている場合でもユーザがそれを聞きもらしてしまうという問題があった。例えば上に述べた「ひらく」の語尾の「く」が抜けて「ひら」だけになってしまった場合でも、「ひら」の再生音が「ひらく」と聞こえてしまうことがある。本実施例では、このような始末端の確認のミスを少なくするため、音声の始末端位置を音により分り易く提示するようにしている。これにより、音声データの確認が音により容易に且つ確実にこなえるようになるため、学習データの収集が簡単でミスなく行なえ、音声入出力インタフェースの使い勝手の向上と認識精度の向上が実現できる。

【0352】始末端位置を分り易くする方法としては、（方法1）検出された音声区間の前後に白色雑音や正弦波など既知の音を付加して再生する方法、（方法2）始末端位置にクリック音を乗せて再生する方法、（方法3）始端よりも一定時間前から終端よりも一定時間後までの発声全体を再生した後、音声区間のみを再生する方法、などが考えられる。

【0353】上記方法1によれば、先程述べた「ひらく」の例では、「ひら」の後にすぐ別の音が続くため、「く」が抜けていることを容易に聞き取ることができる。上記方法2によれば、「ひら」の後に続いて、クリック音が来るため「く」が抜けていることが分る。また、上記方法3によれば、発声全体と音声区間とを比較して聞くことができるため、「く」の有無を容易に識別

することができる。

【0354】ここで、本実施例による拡張したデータ収集部8の構成を図116に示す。

【0355】データ収集部8は、図116に示すように、第9実施例の図29のデータ収集部8に、音声データ確認部411、データ使用可否入力部413を加え、学習データ収集制御部83を介して音声特徴データを音声特徴データ保存部に送るような構成になっている。すなわち、音声データ確認部411で提示された再生音を聞いて、ユーザがその音声データを辞書作成に使うか否かをデータ使用可否入力部413から指定できるような構成になっている。

【0356】このデータ収集部8の処理の流れを図117に従って説明する。

【0357】まず、初期設定では、ユーザからのデータ収集の指示により、データ収集部8から音声認識システム1に対して学習モード設定要求が出され(ステップ11001)、これを受けて音声認識システムは認識対象語彙をデータ収集部8に送る。データ収集部8では認識対象語彙がユーザに表示される(ステップ11002)。

【0358】ユーザにより学習語彙が選択されると(ステップ11003)、データ収集部8は音声認識システム1に単語音声特徴データと単語音声波形データの送信を要求し(ステップ11004)、選択された語彙を発声のガイドとして発声ガイド表示部415に表示し(ステップ11005)、ユーザに発声を促す。音声認識システム1では発声されたユーザの音声を処理した後、データ収集部8に単語特徴データと波形データを送信する。そして、データ収集部8はそのデータを受信し、内部メモリに一時格納する(ステップ11006)。

【0359】音声波形データは音声データ確認部411に送られ、ユーザがそのデータを確認し、辞書作成に使うか否かを、データ使用可否入力部413により入力する(ステップ11007)。データを使用するとした場合には単語音声特徴データが磁気ディスク上などにファイル出力され(ステップ11008でYESの場合およびステップ11009)、使用しないとした場合にはファイル出力しない(ステップ11008でNOの場合)。

【0360】学習終了時にはユーザがデータ収集終了の指示を入力し、データ収集指示フラグがOFFならば(ステップ11010でYesの場合)、データ収集部8は学習モードの解除を音声認識システム1に要求する(ステップ11012)。音声認識システム1では、それを受けて学習モードを解除する。一方、学習を終了しないときは、データ収集指示フラグを検査し(ステップ11011)、上記ステップ11004以下の処理を繰り返す。データ収集指示フラグは、学習データ収集制御部の中に設定されており、図に示すようなデータ収集ボ

タンにより、ユーザが入力可能とすることができる。

【0361】次に、本実施例の音声データ確認部411の構成を図118に示す。

【0362】音声データ確認部411は、音声データを格納する音声データメモリ421、音声データを加工する音声データ加工部422、加工に用いる付加音を生成する付加音生成部424、加工後の音声データを再生して音にする再生部423から成り、学習データ収集制御部83から音声データと始末端位置に関する情報を受け取って加工後、音として出力する。加工後の音を音声入出力システムに送って音データを再生することにすれば、再西部423はなくても良い。

【0363】次に、図119に従って処理の流れについて説明する。

【0364】まず、学習データ収集制御部83から音声データと始末端情報を受け取り、音声データメモリ421に格納する(ステップ12001、ステップ12101、ステップ12201)。この音声データは、音声区間の前後に一定時間、例えば240msecの余裕を付けた波形データであり、例えば図120に示すようなものである。図のデータは「ひらく」の「ひら」が音声区間として検出されたため、「く」の音は終端の余裕の中に入っている。

【0365】次に、音声区間の前後に付加音をつける上記方法1の場合では、付加音を付加音生成部424で作成(ステップ12002)、音声データ加工部422で始末端位置の前と終末端位置の後にこの付加音を付加する(ステップ12003、ステップ12004)。この結果、音声データ図121の(a)に示すようなものになる。

【0366】付加音データは白色ノイズでも良いし、正弦波でも良く、これらは乱数発生ルーチンや三角関数のルーチンを使って容易に作成できる。又、録音データを単に読み出すだけでも良い。

【0367】始末端位置にクリック音を付加する上記方法2の場合では、クリック音を付加音生成部424で作成(ステップ12102)、始末端位置に付加する(ステップ12103、ステップ12104)。この結果、音声データは図121の(b)に示すようなものになる。ここでクリック音は短時間、例えば数10msec幅のパルスや三角波等で良い。

【0368】発声の全体と音声区間の両方を再生する上記方法3の場合では、まず、音声区間外の平均パワーを計算し(ステップ12202)、この値が、しきい値、例えば雑音レベル+2dBよりも大きければ(ステップ12203でYESの場合)、音声区間の前後についた余裕と音声区間とを合わせた音声全体を再生する(ステップ12204)。一方、計算した平均パワーがしきい値よりも小さければ(ステップ12203でNOの場合)、音声区間のみ再生する(ステップ12205)。

雑音レベルは音声認識システム1で音声検出のために常時測定しているため(永田、他“ワークステーションにおける音声認識機能の開発”, 電子情報通信学会技術報告, HC9119, pp. 63-70, (1991)、参照)それを用いれば良い。発声全体の再生と音声区間の再生の2回の再生を、発声の毎に行なうのは煩しいため、上述のように音声区間の外の音声パワーが大きいときに、始末端位置を誤った可能性が大きいと見なして、そのときのみ2回の再生を行なうようにすれば、煩しさを軽減できる。

【0369】この場合、図121の(c)に示すように、発声全体の再生音は「ひらく」の全発声が再生されるが、音声区間のみの再生音は「ひら」だけしか再生されないため、続けてこの2つの再生音を聞いて比較することによって「く」が抜けていることを容易に識別できる。

【0370】以上に述べたように、音声データが正しいか否かをユーザが再生音により容易に判断することができ、データを辞書作成に使用するか否かをデータ収集部で直ちに入力することができるため、音声データ収集を簡単に、且つ確実にこなうことができる。

【0371】これにより、誤ったデータを除いて認識辞書を作成することができる。

【0372】

【発明の効果】本発明によれば、各応用プログラムにより音声認識システムに対する音声認識結果の受信の可否を決定できるので、応用プログラムが自分や他の応用プログラムの音声入力に関する制御を自由に行うことができ、柔軟で使いやすい音声認識インターフェースが構築できる。また、音声認識システムがその音声認識結果を同時に複数の応用プログラムに送信できるので、一つの音声入力による操作を同時に複数の応用プログラムに対して行うこともでき、音声入力による計算機の操作性も向上する。さらに音声認識システムが複数の応用プログラムに対する音声認識を行えるので、音声入力対象の明示的な指定をせずに音声認識結果に基づき音声入力を各応用プログラムに振り分けることができ、利用者の負担を軽減できる。

【図面の簡単な説明】

【図1】本発明の一実施例の概略構成を示す図。

【図2】音声認識部の概略構成を示す図。

【図3】音声認識部の他例の概略構成を示す図。

【図4】音声認識部の他例の概略構成を示す図。

【図5】音声認識部の他例の概略構成を示す図。

【図6】応用プログラムの概略構成を示す図。

【図7】構成要素間で伝送されるメッセージを説明する図。

【図8】入力マスクの種類を示す図。

【図9】音声認識インターフェース各部の処理のタイムチャートを示す図。

【図10】応用プログラム管理テーブルを説明する図。

【図11】本発明の第2実施例の概略構成を示す図。

【図12】一般的なウィンドウシステムの画面表示例を示す図。

【図13】応用プログラムの認識語彙を説明する図。

【図14】入力フォーカスの移動に伴う音声認識語彙の変化を説明する図。

【図15】認識語彙の表示例を説明する図。

【図16】マウスの位置により認識語彙を変更する状態を説明する図。

【図17】本発明の第3実施例での応用プログラムの認識語彙を説明する図。

【図18】応用プログラム管理テーブルを説明する図。

【図19】本発明の第4実施例を説明する図。

【図20】本発明の第5実施例の概略構成を示す図。

【図21】メッセージ表示例を示す図。

【図22】ワークステーションなどのマルチウィンドウ環境を示す図。

【図23】本発明の第6実施例での応用プログラム管理テーブルを示す図。

【図24】図23の応用プログラム管理テーブルに基づく表現を説明する図。

【図25】タスク管理プログラム機能の拡張例を示す図。

【図26】本発明の第7実施例での表示例を説明する図。

【図27】同第7実施例での表示例を説明する図。

【図28】本発明の第9実施例の概略構成を示す図。

【図29】学習データ収集部の概略構成を示す図。

【図30】音声認識システムとのメッセージ交換を説明する図。

【図31】音声認識システムのデータ収集時のフローチャートを示す図。

【図32】学習データ収集部のフローチャートを示す図。

【図33】学習語彙ガイド表示部での表示例を示す図。

【図34】学習語彙ガイド表示部での表示例を示す図。

【図35】データ収集時の音声認識インターフェースの処理の流れを示す図。

【図36】本発明の第10実施例の概略構成を示す図。

【図37】辞書作成管理テーブルを示す図。

【図38】辞書作成管理テーブルを示す図。

【図39】辞書作成管理テーブルを示す図。

【図40】辞書作成管理テーブルへの登録手順を説明する図。

【図41】辞書作成の手順を説明する図。

【図42】辞書作成の進行状況の表示例を示す図。

【図43】辞書作成処理の速度表示の例を示す図。

【図44】辞書作成処理の速度表示の例を示す図。

【図45】本発明の第11実施例の概略構成を示す図。

【図46】音声認識自動停止処理を説明する図。
 【図47】本発明の第12実施例を説明する図。
 【図48】同第12実施例を説明する図。
 【図49】本発明の第13実施例を説明する図。
 【図50】本発明の第14実施例の概略構成を示す図。
 【図51】音声合成部の概略構成を示す図。
 【図52】音声出力管理テーブルを説明する図。
 【図53】音声入力に対するメッセージを説明する図。
 【図54】音声出力に対する入力マスクを説明する図。
 【図55】応用プログラム管理テーブルを説明する図。
 【図56】音声出力処理のフローチャートを示す図。
 【図57】音声出力処理のタイムチャートを示す図。
 【図58】音声出力要求処理のフローチャートを示す図。
 【図59】中断処理のある音声データを重畳する際の一例を説明する図。
 【図60】本発明の第15実施例の概略構成を示す図。
 【図61】応用プログラムと音声入出力システム間で交わされるメッセージを説明する図。
 【図62】音声メールツールが音声データを録音する処理のタイムチャートを示す図。
 【図63】音声メールツールの画面表示例を示す図。
 【図64】音声データ編集用のサブウィンドウを示す図。
 【図65】メール送信による返信の文面例を示す図。
 【図66】音声データ編集用のサブウィンドウを示す図。
 【図67】合成音声の属性のデータベースの一例を示す図。
 【図68】メール読み上げ時に使用する音声コマンドの一例を示す図。
 【図69】音声メールシステムの概略構成を示す図。
 【図70】応用プログラム管理テーブルを説明する図。
 【図71】メールシステムと音声入出力システム間のメッセージを説明する図。
 【図72】タスク重要度管理テーブルを説明する図。
 【図73】音声メールシステムの電子メール処理のフローチャートを示す図。
 【図74】受信メールの通知例を示す図。
 【図75】タスク重要度管理テーブルを説明する図。
 【図76】制御コード交じりのメール例を示す図。
 【図77】本発明の第16実施例の概略構成を示す図。
 【図78】本発明の第16実施例の概略構成を示す図。
 【図79】要約設定処理のフローチャートを示す図。
 【図80】本発明の第17実施例の概略構成を示す図。
 【図81】音声を使ったメール文書作成例を示す図。
 【図82】応用プログラムと音声認識システムの間でのメッセージ例を示す図。
 【図83】音声区間データを入力音声から切り出す処理のタイムチャートを示す図。

【図84】音声によるメール題の入力を説明する図。
 【図85】定型的なメール文書の入力を説明する図。
 【図86】メールアドレスブックの画面表示例を示す図。
 【図87】音声入力可能なメールアドレスの登録例を示す図。
 【図88】音声によるメール送付先指定の手順を説明する図。
 【図89】メールアドレスのデータベースを用いたメール送付先指定を説明する図。
 【図90】本発明の第18実施例の概略構成を示す図。
 【図91】同第18実施例におけるシステム構成を示す図。
 【図92】同第18実施例での画面表示例を示す図。
 【図93】音声インターフェース管理テーブルの一例を示す図。
 【図94】疑似音声フォーカスと音声フォーカスとの対応関係を示す図。
 【図95】メッセージ変換部のフローチャートを示す図。
 【図96】本発明の第19実施例の概略構成を示す図。
 【図97】同第19実施例での画面表示例を示す図。
 【図98】同第19実施例のより詳細な構成を示す図。
 【図99】音声インターフェース管理テーブルの一例を示す図。
 【図100】音声フォーカスの表示方法を説明するための図。
 【図101】外付けウィンドウの表示例を示す図。
 【図102】応用プログラム管理テーブルの一例を示す図。
 【図103】音声入出力システムの認識処理のフローチャートを示す図。
 【図104】本発明の第20実施例の概略構成を示す図。
 【図105】プログラム操作の登録画面の一例を示す図。
 【図106】プログラム操作登録の処理手順を示す図。
 【図107】本発明の第20実施例の概略構成を示す図。
 【図108】認識辞書の構成の一例を示す図。
 【図109】辞書編集部の概略構成を示す図。
 【図110】辞書編集画面の一例を示す図。
 【図111】辞書編集部の処理のフローチャートを示す図。
 【図112】本発明の第22実施例の概略構成を示す図。
 【図113】応答音声管理部の概略構成を示す図。
 【図114】メッセージ変換部の処理のフローチャートを示す図。
 【図115】応答音声登録部の概略構成を示す図。

【図116】拡張したデータ収集部の概略構成を示す図。

【図117】図116のデータ収集部の処理のフローチャートを示す図。

【図118】音声データ確認部の概略構成を示す図。

【図119】音声データ確認部の処理のフローチャートを示す図。

【図120】音声データの一例を示す図。

【図121】加工後の音声データの様子を示す図。

【図122】従来の音声認識インターフェースを示す図。

【図123】従来の音声認識インターフェースを示す図。

【図124】従来の音声認識インターフェースを示す図。

【図125】従来の音声認識インターフェースを示す図。

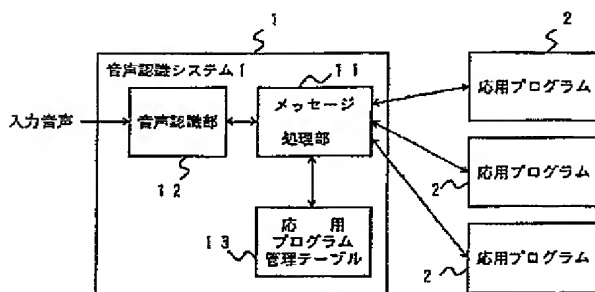
【図126】従来の音声認識インターフェースを示す図。

【符号の説明】

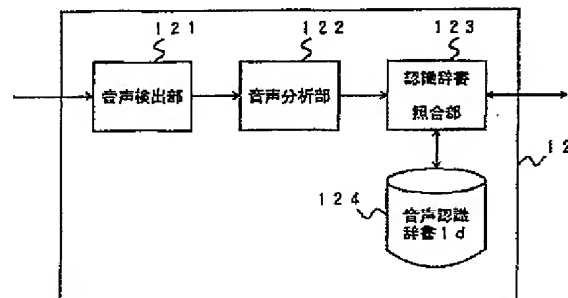
1、3、6…音声認識システム、11…メッセージ処理部、12…音声認識部、121…音声検出部、122…音声分析部、123…認識辞書照合部、124…音声認識辞書、13…応用プログラム管理テーブル、2、5、7…応用プログラム、21、71…メッセージ入出力部、22…プログラム本体、4…ウィンドウシステム、8…データ収集部、81…単語音声特徴データ保持部、82…学習語彙表示選択部、83…学習データ収集制御*

*部、84…学習語彙ガイド表示部、9…辞書作成部、91…辞書作成管理部、92…辞書作成制御部、93…データ入力部、94…辞書作成部本体、95…ファイル出力部、10…音声認識自動停止部、14…音声合成部、561…全体制御部、562…波形重置部、563…音声出力管理テーブル、564…波形合成部、651…音声入出力システム、652…ウィンドウシステム、653…音声メールツール、6531…電子メール処理部、6532…メッセージ入出力部、821…音声入出力システム、822…音声メールシステム、8221…電子メール処理部、8222…文書要約部、8223…メッセージ入出力部、851…音声認識システム、852…音声メールシステム852、853…メールアドレステーブル、103…汎用応用プログラム（GAP）、102…専用応用プログラム（SAP）、104…音声インターフェース管理システム（SIM）、141…音声インターフェース管理部、142…プログラム操作登録部、143…メッセージ変換部、23…音声ウィンドウ、1440₀…1440₀…音声ウィンドウ、151…プログラム操作表示編集部、152…登録内容保存部、153…ウィンドウID取得部、144…辞書編集部、441…辞書内容表示編集部、442…辞書内容検索部、401…応答音声管理部401、403…応答音声登録部、411…音声データ確認部、413…データ使用可否入力部、415…発生ガイド表示部、421…音声データメモリ、422…音声データ加工部、423…再生部、424…付加音データ保存部。

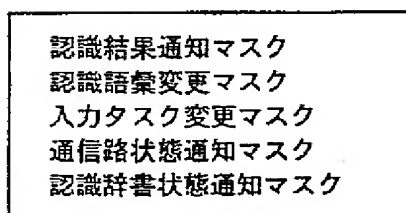
【図1】



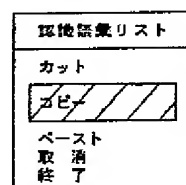
【図2】



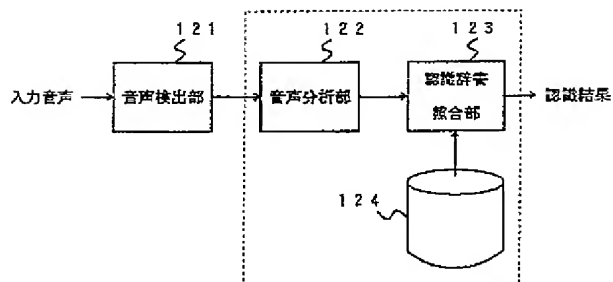
【図8】



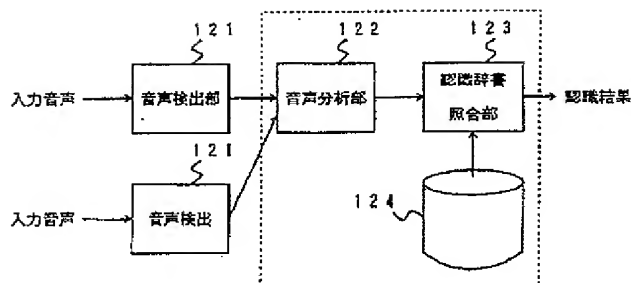
【図15】



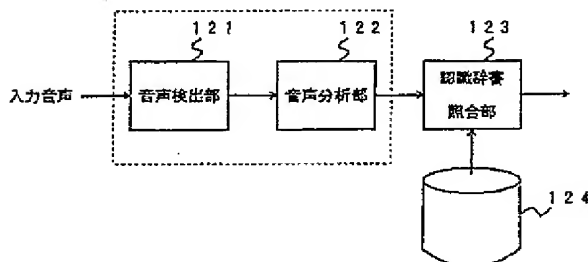
【図3】



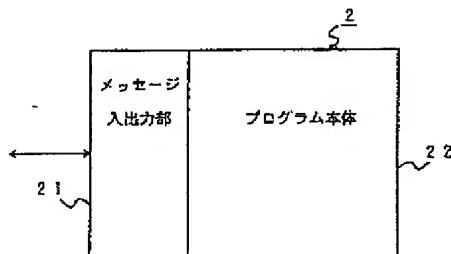
【図4】



【図5】



【図6】



【図7】

	発信	受信	メッセージ
(a)	応用プログラム	メッセージ処理部	通信路接続/切断要求 音声認識辞書のロード/解放要求 認識語彙設定要求 入力タスク設定要求 入力マスク設定要求 認識開始/終了要求
(b)	メッセージ処理部	応用プログラム	音声認識結果 入力タスク変更通知 通信路接続/切断の通知 認識語彙変更通知 認識辞書ロード/解放通知
(c)	音声認識部	メッセージ処理部	認識語彙照会要求 音声認識結果
(d)	メッセージ処理部	音声認識部	認識辞書ロード/解放要求 認識語彙情報

【図13】

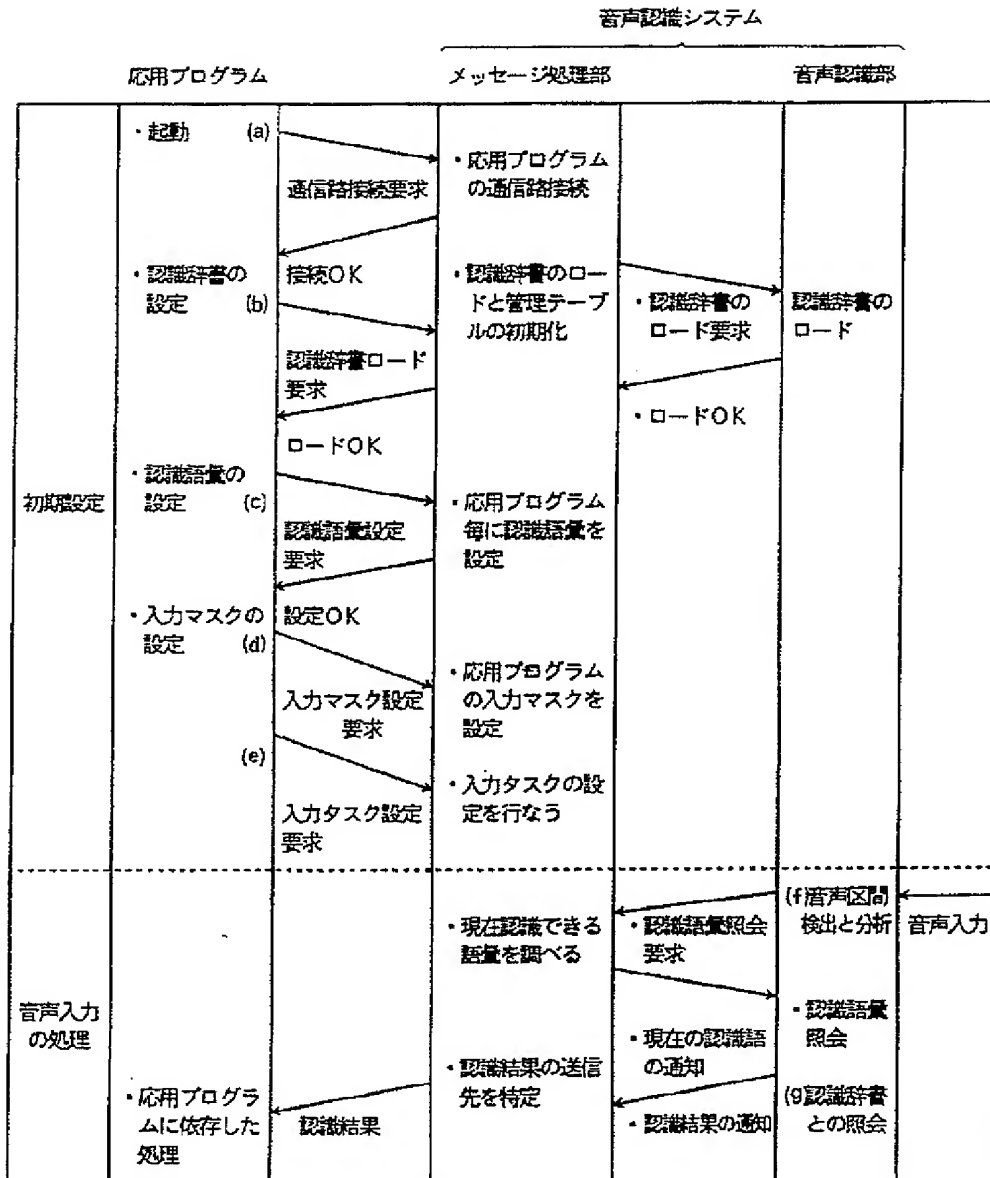
応用プログラム名	認識語彙
シェルツール	終了 ヒストリー プロセス リスト ホーム
テキストエディタ	カット コピー ペースト 終了 取消
メールツール	先頭 最後 前次 終了 送信

【図21】

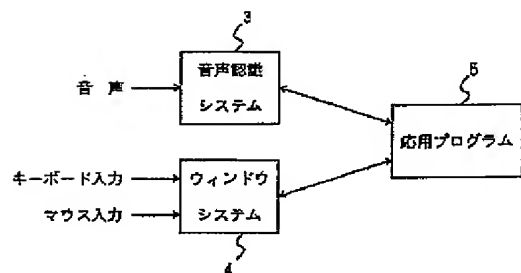
新しいメールを受信しました
今すぐお読みになりますか？

はい いいえ

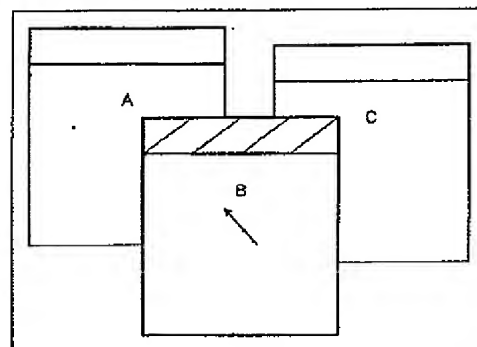
【図 9】



【図 11】



【図 12】



【図10】

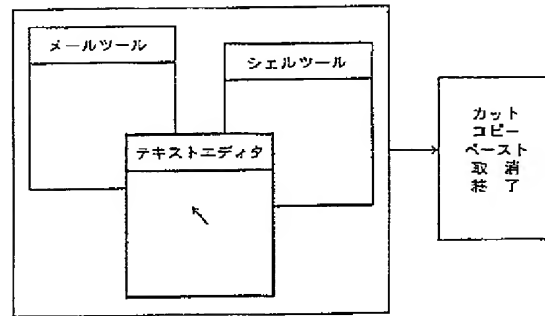
プログラムID	入力マスク	音声入力フラグ	認識対象語彙リスト
0 (シェルツール)	認識結果 通知マスク 入力タスク変更 マスク	1	ヒストリー、プロセス、 リスト、ホーム、終了
1 (テキストエディタ)	認識結果 通知マスク	0	カット、コピー、 ペースト、取消、終了

(a)

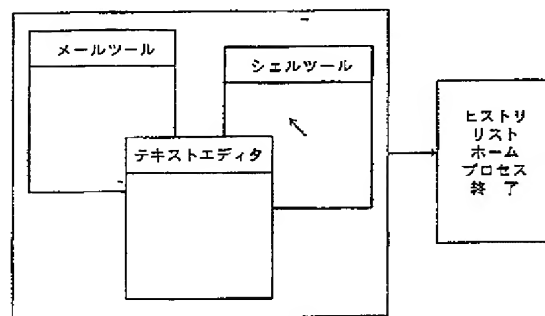
プログラムID	入力マスク	音声入力フラグ	認識対象語彙リスト
0 (シェルツール)	認識結果 通知マスク 入力タスク 変更マスク	0	ヒストリー、プロセス、 リスト、ホーム、終了
1 (テキストエディタ)	認識結果 通知マスク	0	カット、コピー、 ペースト、取消、終了
2 (メールツール)	認識結果 通知マスク	1	先頭、最後、前、次、 送信、終了

(b)

【図14】

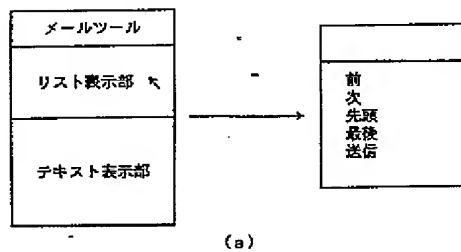


(a)

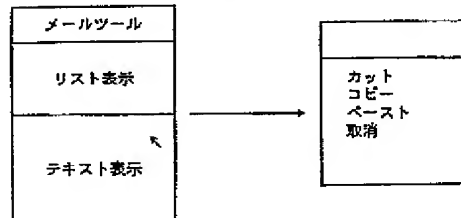


(b)

【図16】



(a)



(b)

【図17】

応用プログラム名	認識語彙	
	ローカル	グローバル
シェルツール	リスト ヒストリ ホーム プロセス 終了	シェルツール
テキストエディタ	カット コピー ペースト 終了 取消	テキストエディタ
メールツール	先頭 最後 前次 送信	メールツール

【図25】

タスク管理		
排他制御	全部	反転
メールツール		
テキストエディタ		
シェルツール		

【図33】

【図26】

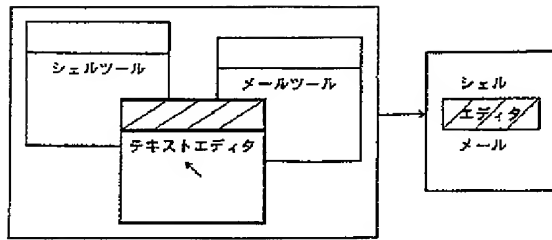
認識対象語彙	
メールツール	シェルツール
先頭 最後 前 次 終了 返事	エルエス クリア ホーム ヒストリ プロセス 終了

【図18】

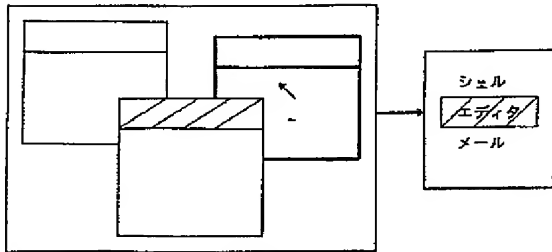
プログラムID	入力マスク	音声入力 フラグ	認識対象 語彙リスト	ローカル	グローバル
0 (シェル-#)	認識結果 通知マスク 入力タスク 変更マスク	0	ヒストリー プロセス リスト ホーム 終了 シェル-#	○ ○ ○ ○ ○ ×	× × × × × ○
1 (テキストディ)	認識結果 通知マスク	0	カット コピー ペースト 取消 終了 テキストディ	○ ○ ○ ○ ○ ×	× × × × × ○
2 (メール-#)	認識結果 通知マスク	1	先頭 最後 前次 送信 終了 メール-#	○ ○ ○ ○ ○ ×	× × × × × ○

取消
カット
コピー
ペースト
フォント

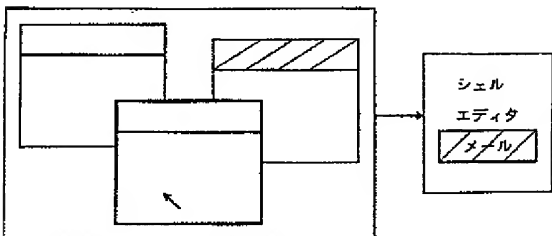
【図19】



(a)

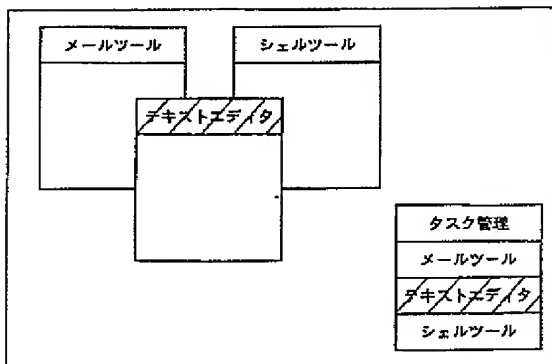


(b)



(c)

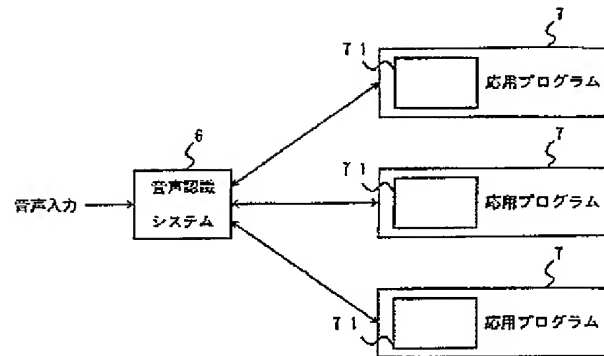
【図22】



【図27】

タスク管理		
メールツール	シェルツール	テキストエディタ
先頭	エクス	カット
最後	クリア	コピー
前次	ホーム	ペースト
終了	履歴	取消
返事	終了	終了

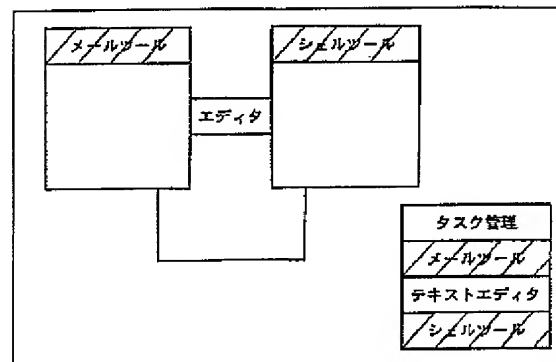
【図20】



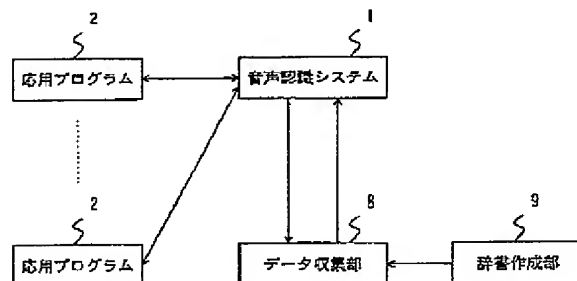
【図23】

プログラムID	入力マスク	音声入力フラグ	認識対象語彙リスト
0 (シェル-4)	認識結果 通知マスク 入力タスク 変更マスク	1	ヒストリー、プロセス、 リスト、ホーム、終了
1 (テキスト)	認識結果 通知マスク	0	カット、コピー、 ペースト、取消、終了
2 (メール-6)	認識結果 通知マスク	1	先頭、最後、前、次、 送信、終了

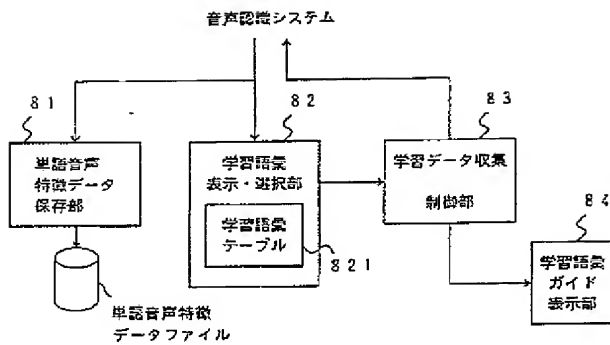
【図24】



【図28】



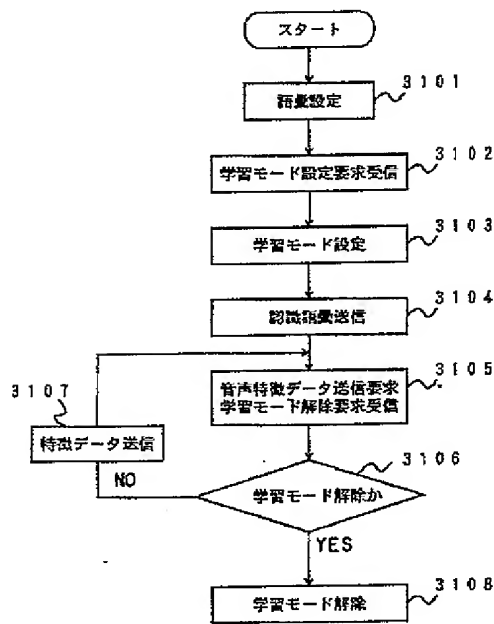
【図29】



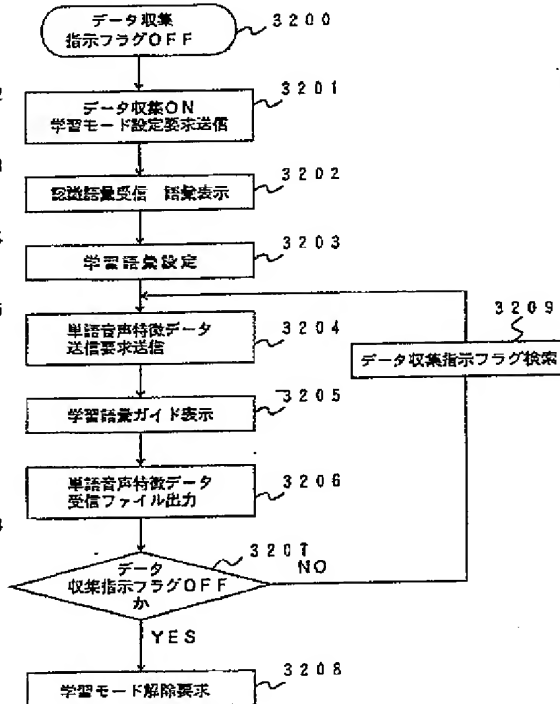
【図30】

発信	受信	メッセージ
学習データ収集部	音声認識システム	学習モード設定要求 学習モード解除要求 単語音声特徴データ要求
音声認識システム	学習データ収集部	認識語彙 単語音声特徴データ

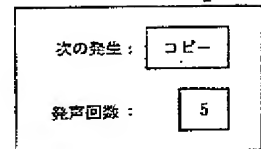
【図31】



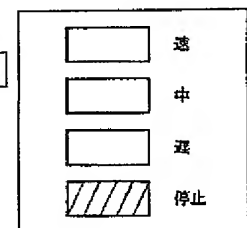
【図32】



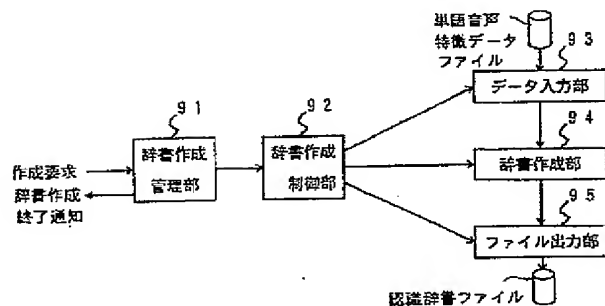
【図34】



【図43】



【図36】



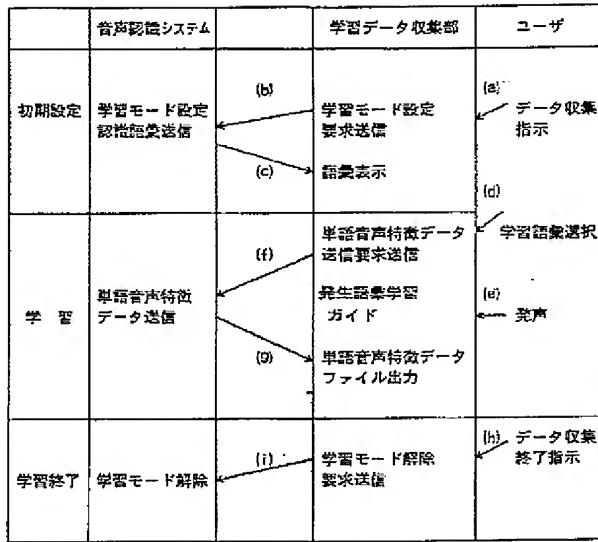
【図37】

要求日時	語彙
92.9.1.11.3	コピー
92.9.1.11.5	ペースト
92.9.2.22.10	カット
	:

→ 作成中

← 新しい要求カット

【図35】



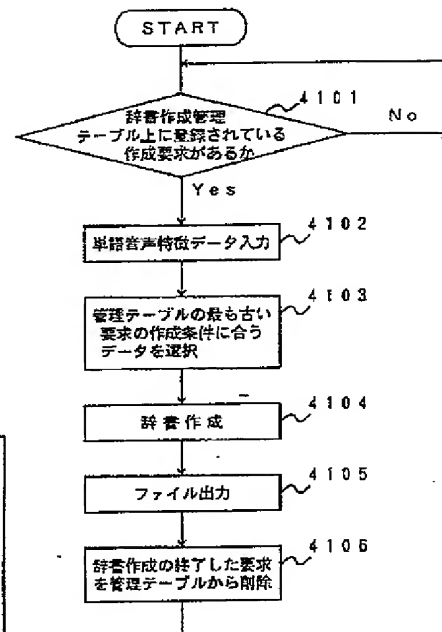
【図38】

要求日時	語彙	ユーザ名
92.9.1.11.2	右	A
92.9.1.11.3	左	B
92.9.2.22.3	右	B
	:	:

【図54】

- ・音声出力完了通知マスク
- ・優先度設定通知マスク
- ・合成音声データ通知マスク

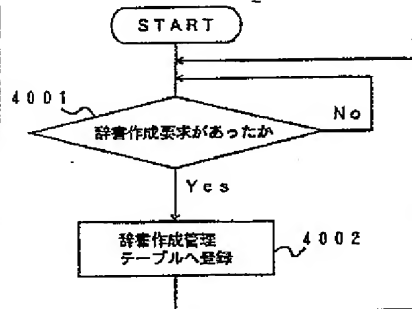
【図41】



【図39】

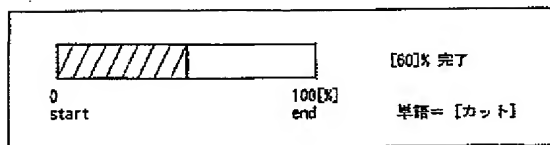
要求日時	語彙	ユーザ名	日付指定
92.9.1.11.4	上	A	92.1.20より後
92.9.1.11.23	下	A	92.1.20より後
92.9.3.13.55	削除	A	90.11.1より後

【図40】



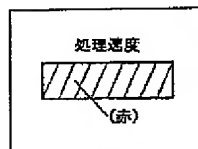
【図68】

【図42】



(a)

【図44】



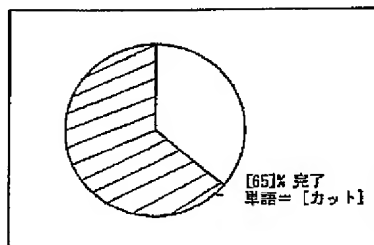
(a)

速度	色
速	青
中	緑
遅	黄
停止	赤

(b)

- ・次のメール
- ・前のメール
- ・次の段落
- ・前の段落
- ・次の文
- ・前の文
- ・ストップ
- ・もう1度
- ・次
- ・前
- ・移行
- ・速く
- ・ゆっくり
- ・高く
- ・低く
- ・男性
- ・女性

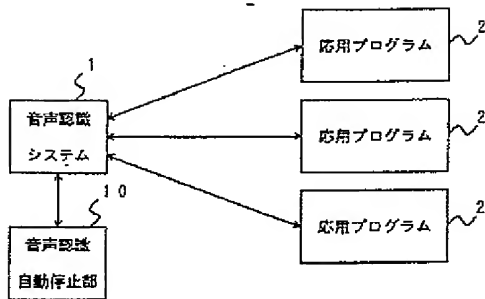
【図61】



(b)

発信	受信	メッセージ
応用プログラム	音声入出力システム	録音開始/停止要求 再生開始/停止要求 音声データ要求 音声データ解放要求
音声入出力 システム	応用プログラム	録音終了 再生終了

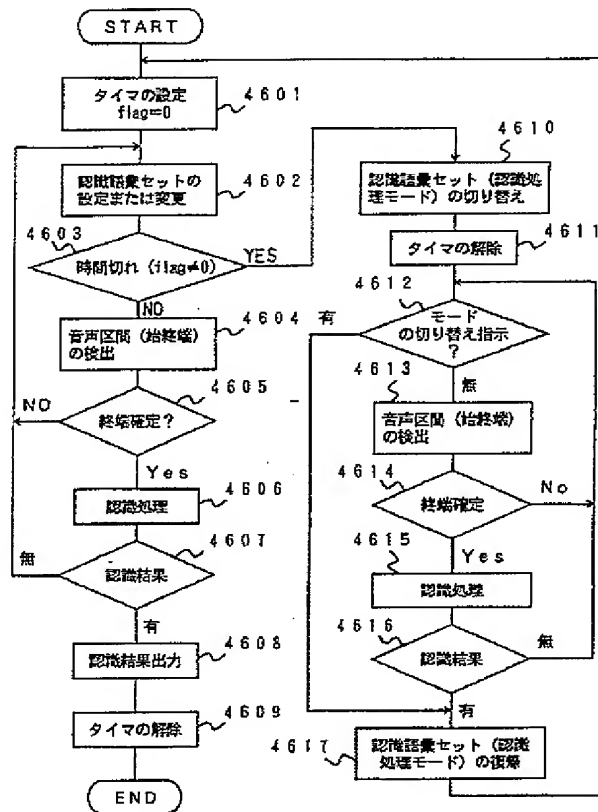
【図45】



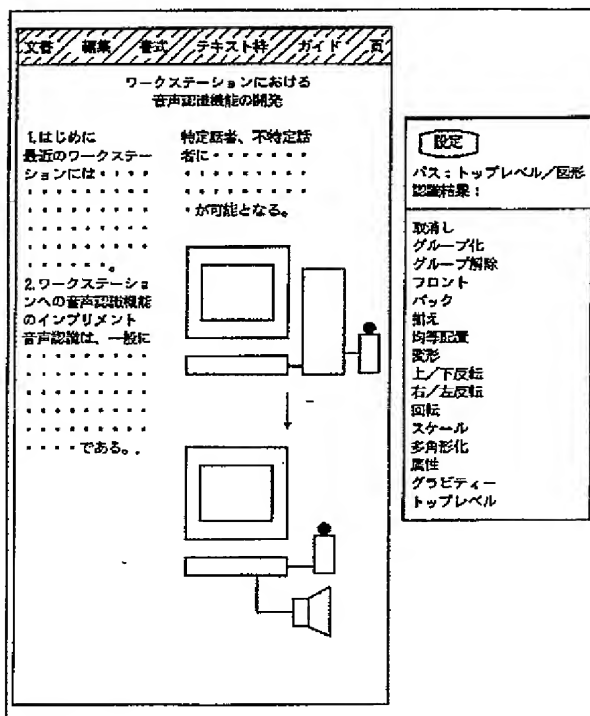
【図72】

応用プログラム	タスク優先度	音声フォーカス
シェルツール	2	1
DTPシステム	5	0
メールシステム	10	0

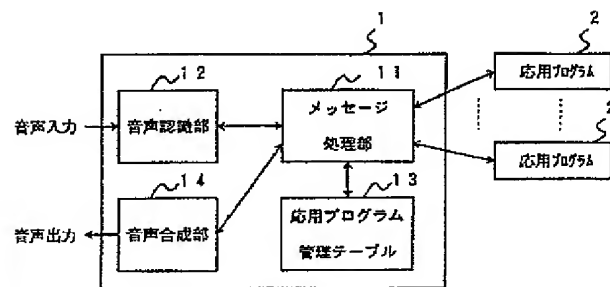
【図46】



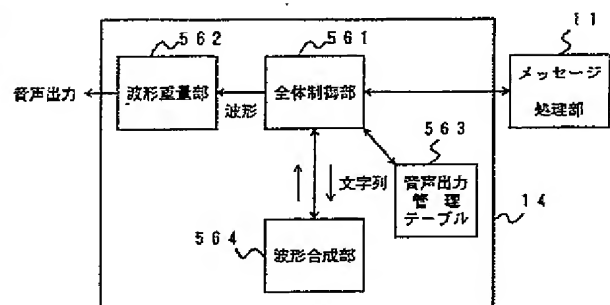
【図49】



【図50】



【図51】



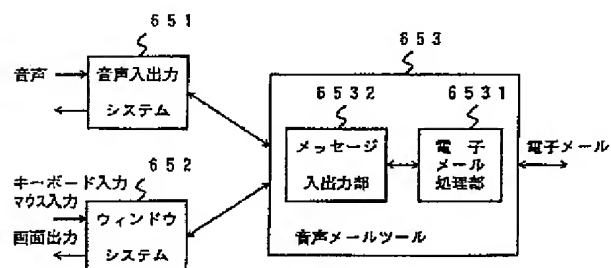
【図47】

<div style="display: flex; justify-content: space-around; margin-bottom: 5px;"> ファイル メール 編集 リスト 表示 </div> <p>表示階層 : _____</p>	<div style="border: 1px solid black; padding: 2px; text-align: center;">認識語彙</div> <p> オープン クローズ 終了 アンドゥ 前 カット ペースト 今日 機能 先頭 最後 引用 サイン 返事 送信 シェルツール フロント バック アイコン化 メールツール 削除 全選択 同僚 上司 連絡会 会議通知 緊急 エディタ システム </p>
<p>1 kishimoto Tue Sep 15 11:24 20/427 tokkyo no kaigi</p> <p>2 nakayama Tue Sep 15 18:59 10/243 meeting</p> <p>3 kishimoto Wed Sep 16 16:17 15/390 kenkyukai</p> <p>4 karasawa Wed Sep 16 22:44 16/365 sadamoto-san</p>	<div style="border: 1px solid black; padding: 2px; text-align: center;">クライアント</div> <p> シェルツール システム エディタ メールツール </p>
<p>From nakayama tue Sep 15 18:59:31 1992</p> <p>To: sawada, nakayama, karasawa, kishimoto, tanaka, nii</p> <p>Subject: meeting</p> <p>Status: 0</p> <p>16日の連絡会は17日 8 : 30からに変更します。 ご参集下さい。</p> <p style="text-align: center;">中山</p>	

【図48】

ファイル	メール	編集	リスト	表示
表示階層: <u>上司</u>				
6 sawada Thu Sep 17 18:20 13/249 gokurousama				
10 sawada Sat Sep 19 19:30 10/173 houkokusho(kinkyu)				
<p>From sawada Sat Sep 19 19:30:49 1992 To: tamura Subject: houkokusho(kinkyu) Status: 0</p> <p>田村殿</p> <p>先週の出張報告書を至急提出のこと。</p> <p>沢田</p>				
<p>To: sawada Subject: Re: houkokusho(kinkyu)</p> <p>沢田様</p> <p>>> 田村殿 >> >> 先週の出張報告書を至急提出のこと。 >> >> 沢田</p> <p>遅れて申し訳ありません。今日中に提出します。</p>				

【図60】



【図65】

>> From : kanji
 >> To : tamura
 >> Subject : Sobetsukai
 >> 送別会のお知らせ
 >>
 >> 3月15日、PM6:30から、4階食堂でDr. G
 >> の送別会が行なわれます。
 >> ぜひ参加して下さい

関西芝 営業 田村 余一



【図52】

データID	データ種類	要求受付時刻	合成音テキスト	処理状態	出力ありなし
#0	波形	(時:分:秒)	—	終了	あり
#1	テキスト	5:13:14	「こんにちは」	終了	なし
#2	テキスト	5:14:00	「緊急です」	終了	あり
#3	テキスト	5:14:02	「メール来よ」	処理中	あり
#4	波形	5:14:03	—	未処理	あり
...		5:14:04			

(a)

データID	音量(0~10)	出力中断	合成処理優先度(0~10)	排他処理係数
#0	3	なし	—	—
#1	—	—	—	—
#2	10	あり	10	0.2
#3	7	なし	5	—
#4	3	なし	—	—
...				

(b)

【図53】

発信	受信	メッセージ
a 応用プログラム	メッセージ処理部	<ul style="list-style-type: none"> 音声合成要求 (TEXT→波形) 波形再生要求 音声合成再生要求 優先度設定要求 (音声レベル/時間) 音声出力状況通知要求 合成用パラメータ設定要求 音声波形データ要求
b メッセージ処理部	応用プログラム	<ul style="list-style-type: none"> 合成音波形データ 音声出力状況通知 優先度設定通知
c 音声合成部	メッセージ処理部	<ul style="list-style-type: none"> 音声出力状況通知 優先度設定通知 合成音波形データ
d メッセージ処理部	音声合成部	<ul style="list-style-type: none"> 音声合成要求 (含テキストデータ) 波形再生要求 音声合成再生要求 優先度設定要求 (音声レベル/時間) 音声出力状況通知要求 音声波形データ要求

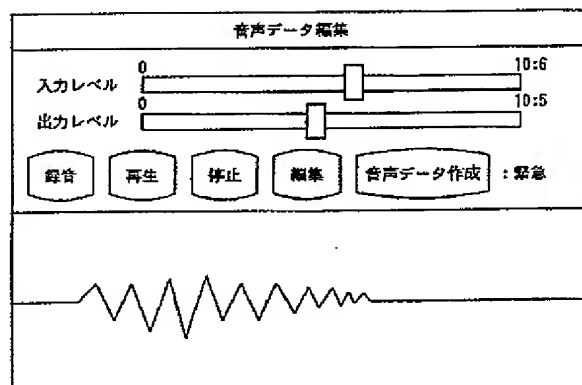
【図75】

【図55】

プログラムID	入力マスク	音声入力フラグ	認識対象語彙リスト	音声出力優先度		
				レベル(1~10)	音声合成処理(1~10)	中断処理(0/1)
0 (システム)	認識結果通知マスク 入力マスク 変更マスク	0	ヒストリー、プロセス、リスト、ホーム、終了	5	5	0
1 (システム)	認識結果通知マスク	0	カット、コピー、ペースト、取消、終了	2	5	0
2 (システム)	認識結果通知マスク	1	先取、最後、前、次、送信、終了	10	8	0

応用プログラム	タスク優先度	音声フォーカス
シェルツール	2	0
DTPシステム	5	1
メールシステム	10	0

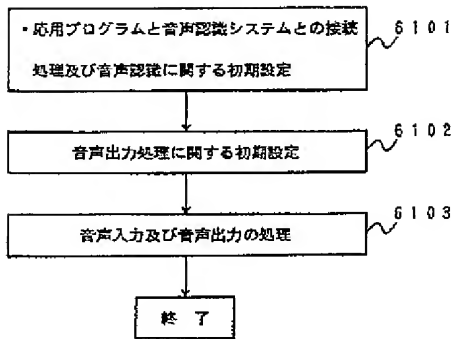
【図64】



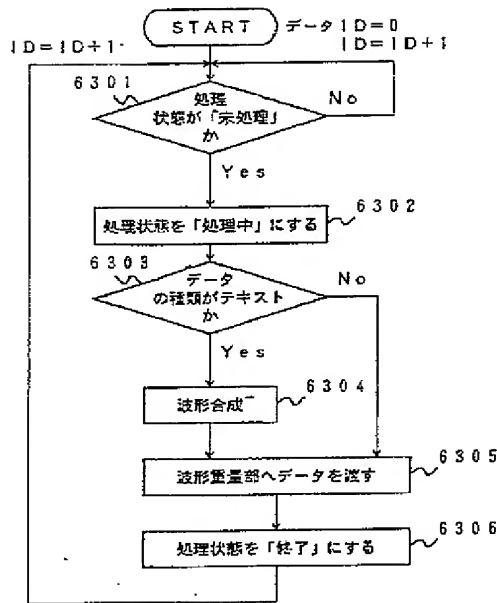
【図67】

発信者のメールアドレス	性別	声の高さ (1~10)	速度(1~10)	大きさ (1~10)
tamura@abc.def.co.jp	男	2	4	5
nakayama@ghi.jkl.co.jp	女	5	8	5
*	男	5	5	5

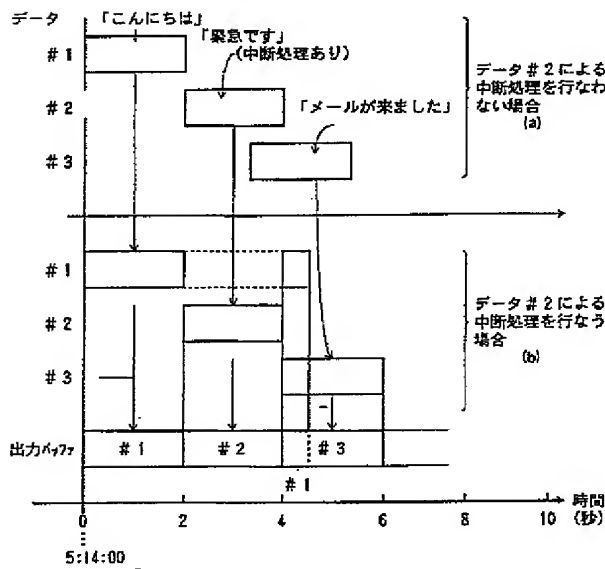
【図56】



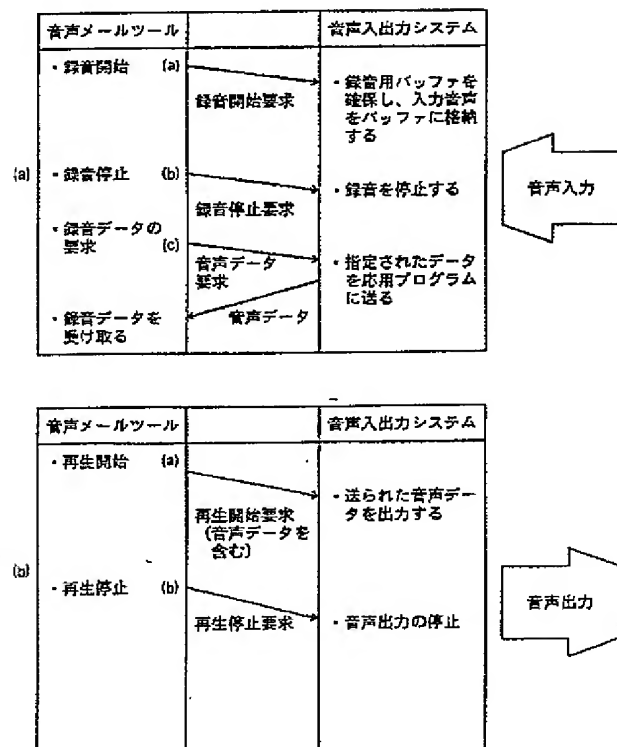
【図58】



【図59】



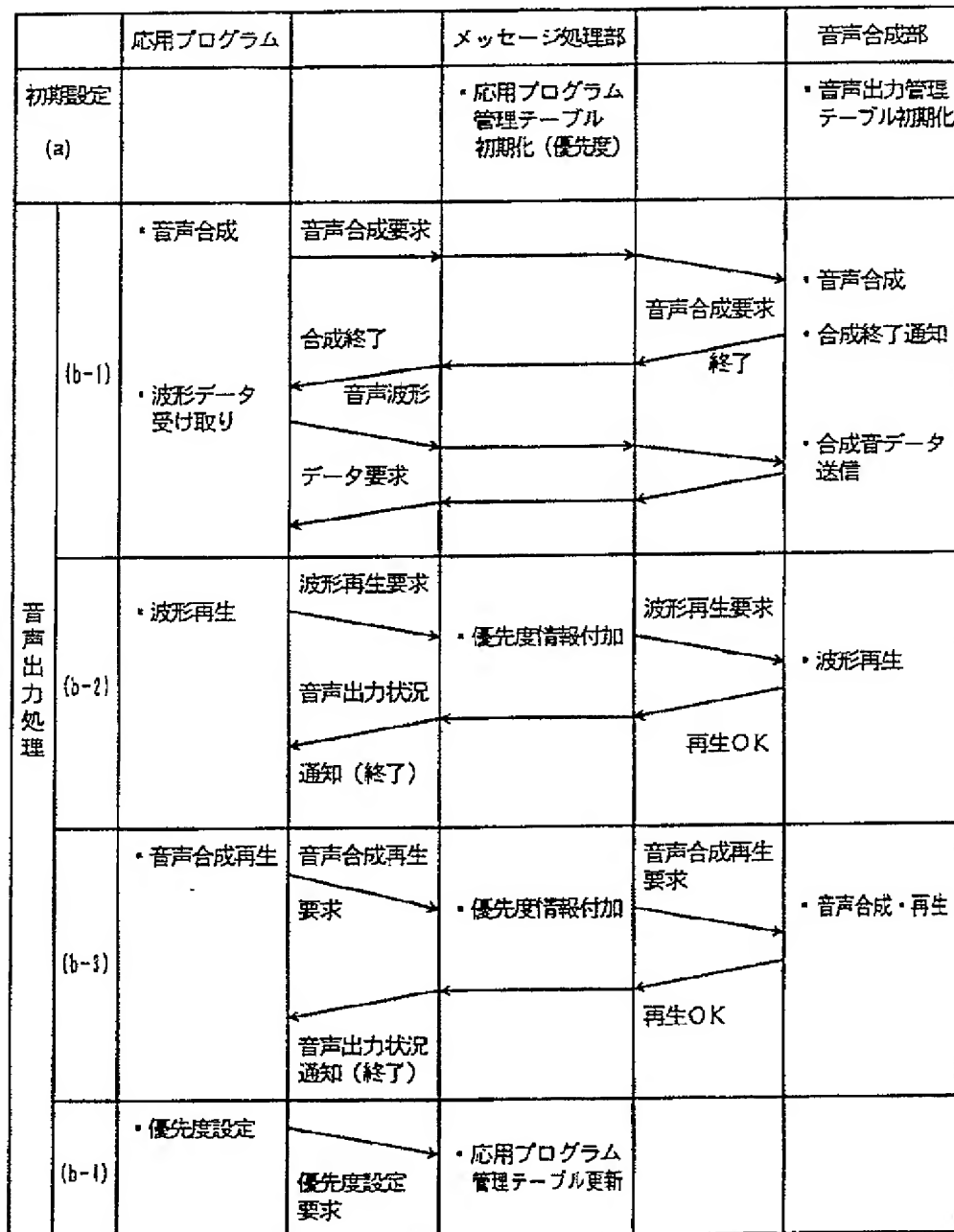
【図62】



【図71】

発信	受信	メッセージ
メールシステム (応用プログラム)	音声入出力システム	タスク優先度設定要求 " 取得 "
音声入出力システム	メールシステム (応用プログラム)	タスク優先度変更通知

【図57】



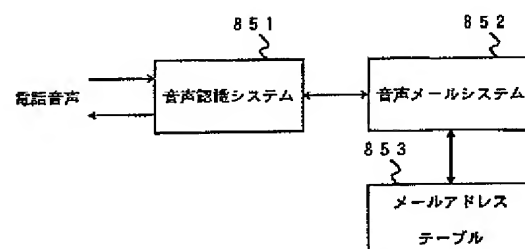
【図76】

Subject : 会議通知

今週の連絡会は水曜日の8:30に変更
します。重要な話しがありますので、
@<male. 5, 5, 9 : 絶対に遅れないように>
して下さい。

以 上

【図80】



【図63】


ファイル メール 編集 リスト 表示 送上げ 全文
表示階層：上司

6 sawada Thu Sep 17 18:20 13/249 gokurousana

☆10 sawada Sat Sep 19 9:30 10/173 houkokusho(kinkyu)

From sawada Sat Sep 19 19:30:49 1992
To: tamura
Subject: houkokusho(kinkyu)
Status: 0

田村殿

先週の出張報告書を至急提出のこと。  緊急

沢田

To: sawada
Subject: Re: houkokusho(kinkyu)

沢田様

>> 田村殿
>>
>> 先週の出張報告書を至急提出のこと。
>>
>> 沢田

遅れて申し訳ありません。今日中に提出します。

【図66】

ファイル メール 編集 リスト 表示 送上げ 全文
表示階層：上司

6 sawada Thu Sep 17 18:20 13/249 gokurousana

☆10 sawada Sat Sep 19 19:30 10/173 houkokusho(kinkyu)

From sawada Sat Sep 19 19:30:49 1992
To: tamura
Subject: houkokusho(kinkyu)
Status: 0

田村殿

先週の出張報告書を至急提出のこと。


沢田

To: sawada
Subject: Re: houkokusho(kinkyu)

沢田様

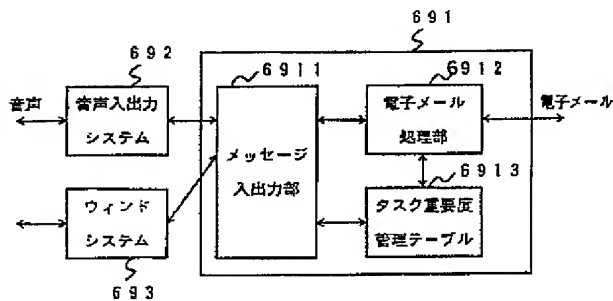
>> 田村殿
>>
>> 先週の出張報告書を至急提出のこと。
>>
>> 沢田

遅れて申し訳ありません。今日中に提出します。

 了解

再生
停止
録音
再生
停止
新規 : 了解
削除

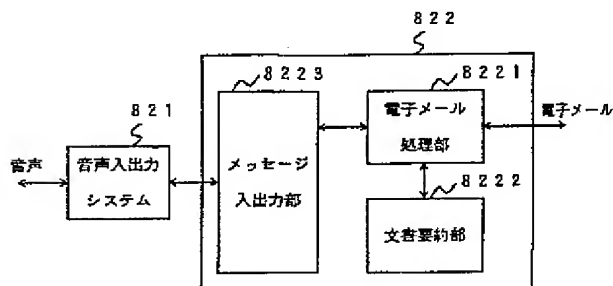
【図69】



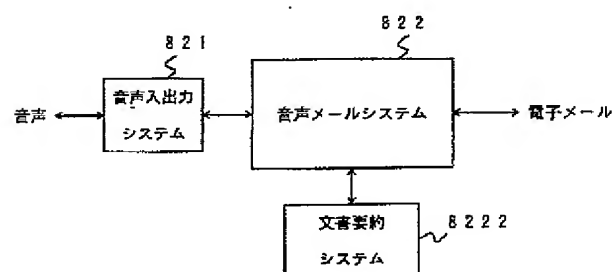
【図70】

プログラムID	入力マスク	音声入力 フラグ	認識対象語彙リスト	タスク 優先度
0 (スタート)	認識結果 通知マスク 入力タスク 変更マスク	0	ヒストリー、プロセス、 リスト、ホーム、終了	2
1 (スタート)	認識結果 通知マスク	0	カット、コピー、 ペースト、取消、終了	5
2 (スタート)	認識結果 通知マスク	1	先頭、最後、前、次、 送信、終了	3

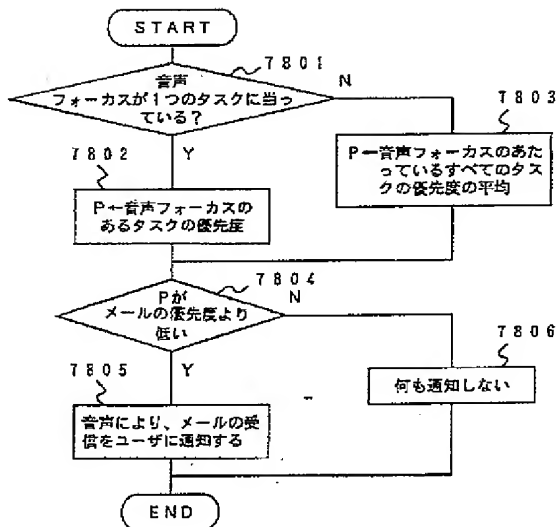
【図77】



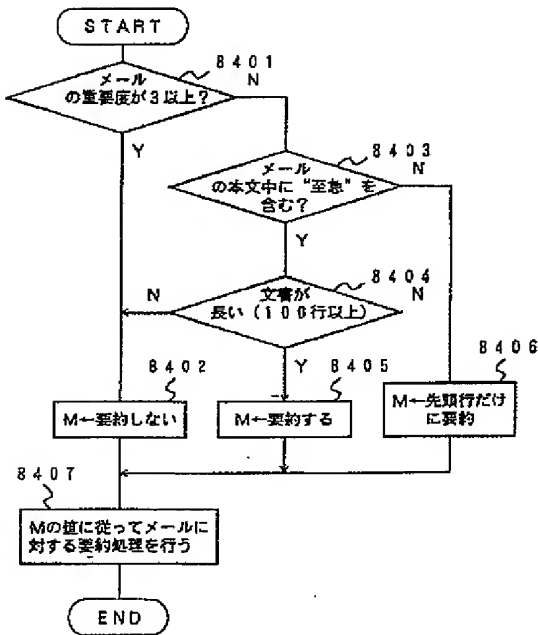
【図78】



【図73】



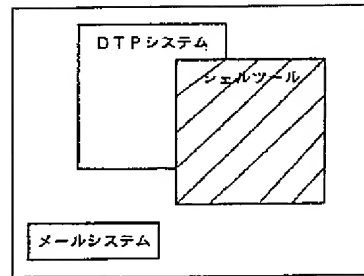
【図79】



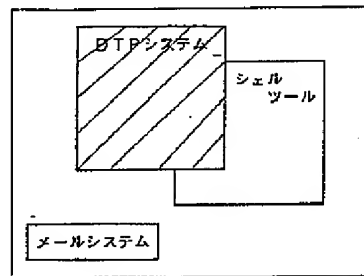
【図82】

発信	受信	メッセージ
応用プログラム	音声認識システム	音声区間検出要求

【図74】



(a)

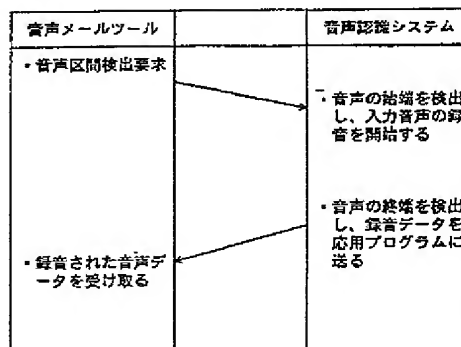


(b)

【図81】

- (1) ユーザ「録音開始」
- (2) ユーザ「〇〇です。××の件ですが、私は都合により行くことができません。申しわけありません。皆様によりしくお伝え下さい。ストップ、ストップ」
- (3) ユーザ「内容確認」
- (4) システム「〇〇です。～お伝え下さい」
- (5) ユーザ「送信」
- (6) システム「△△さんへのメールの送信を完了しました」

【図83】

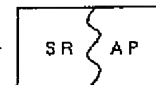


音声入力

【図94】

音声フォーカス	音声フォーカス
シェルツール	SIM
エディタ	SIM
メールツール	メールツール
SIM	SIM

【図122】



【図84】

- (1) ユーザ「サブジェクト」
 (2) ユーザ「会議通知」
 (3) システム「メールのサブジェクトは会議通知」です

【図85】

Subject: ごくろうさま
 \$ {receiver} 様、ごくろうさまでした
 今後ともよろしくお願いします。
 \$ {sender}

【図86】

メールアドレスブック			
登録	削除	変更	
鈴木一郎	suzuki @aaa, bbb, ccc, co.jp		○
田中太郎	tanaka @ddd, eee, fff, ac.jp		—

【図87】

メールアドレスの登録

名前: 鈴木一郎
 アドレス: suzuki @aaa, bbb, ccc, co.jp
 発声回数: 2

OK キャンセル

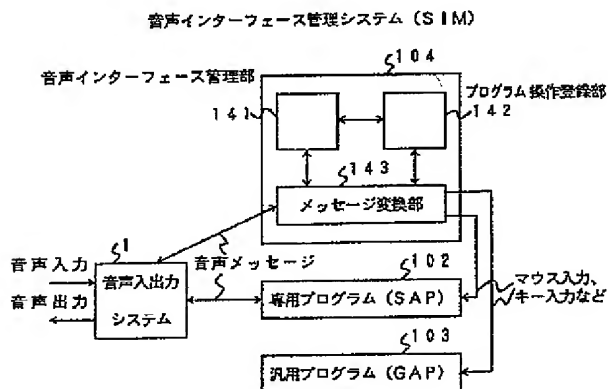
【図89】

【図88】

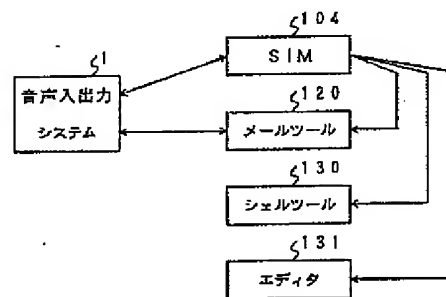
- (1) ユーザ「送信先」
 (2) メールシステム「音声で指定できるメールアドレスは5件あります」
 (3) ユーザ「鈴木」
 (4) メールシステム「送信先は、鈴木、suzuki @aaa, bbb, ccc, co.jpですね」
 (5) ユーザ「はい」

システム「ノードj pは2つのノードを持っています」
 “「1, aaaと、2, bbbです」
 ユーザ「1」
 システム「ノードaaaは、3つのノードを持っています」
 :

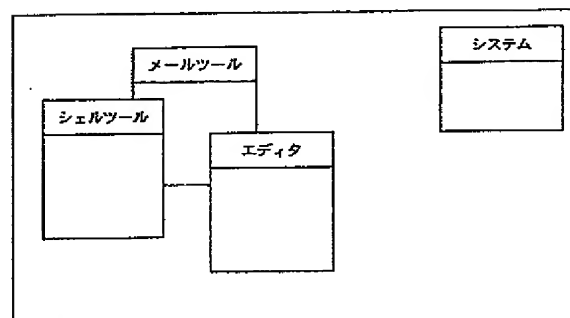
【図90】



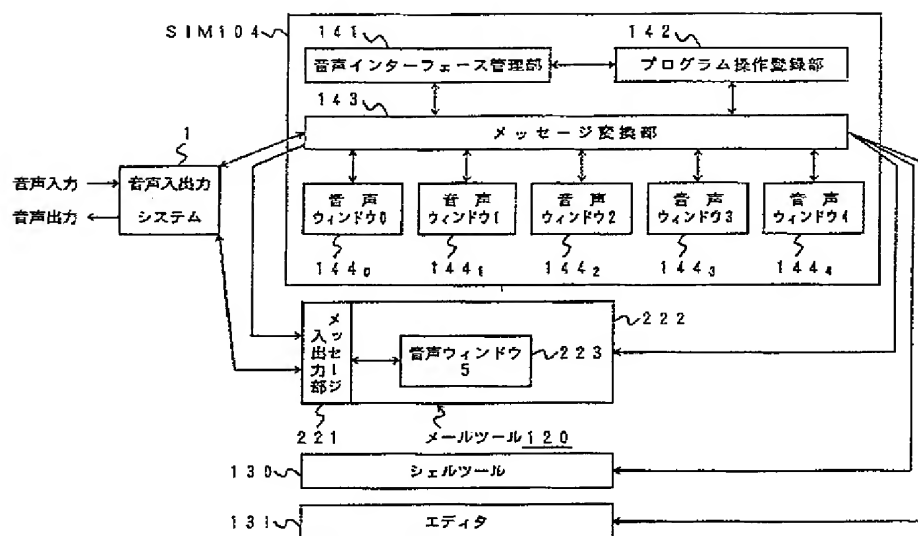
【図91】



【図92】



【図98】

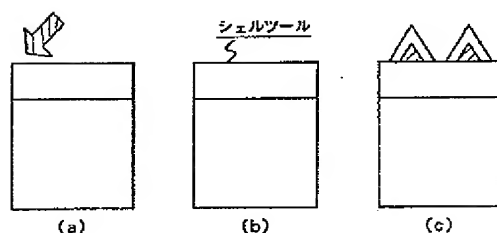


【図99】

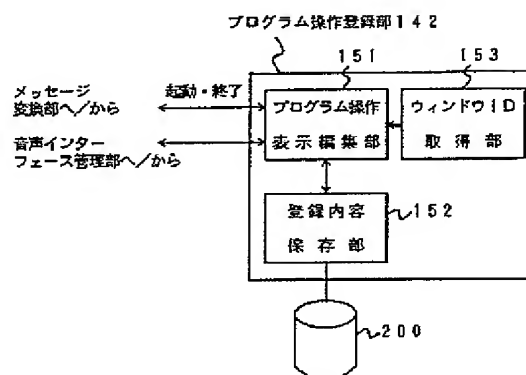
応用プログラム	ウィンドウ ID	音声ウィンドウ ID	ウィンドウ名	疑似音声フォーカス	コマンド名	コマンド
シェルツール	101	1	シェルツール(2)	○	エルエス(0)	key ("Es\n")
	102	2	設定(1)	×	プロセス(0)	key ("Es\n")
					オーケー(0)	press("OK")
エディタ	103	3	エディタ(2)	×	キャンセル(0)	press("CANCEL")
					カット(0)	key ("^X")
					コピー(0)	key ("^C")
	104	4	設定(1)	×	ペースト(0)	key ("^P")
					オーケー(0)	press("OK")
SIM	100	0	システム(2)	×	キャンセル(0)	press("CANCEL")
メールツール	105	5	メールツール(2)	×	終了(0)	press("QUIT")

0:ローカル
1:ウィンドウ
2:グローバル

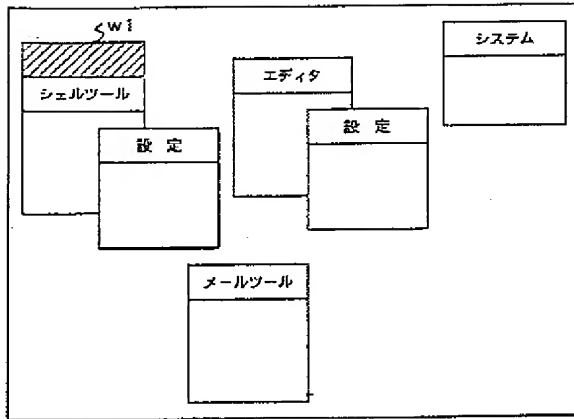
【図101】



【図104】



【図100】



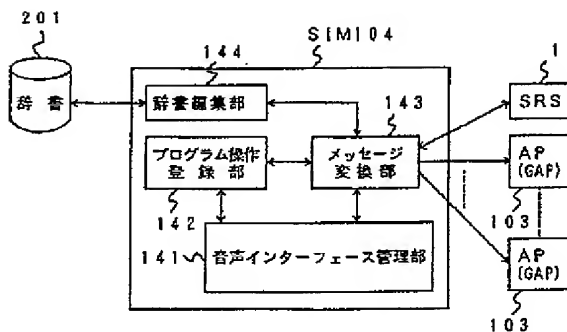
【図105】

登録	取消	終了	ウィンドウID取得	APクラス: シェル ウィンドウ名: シェル1 認識語彙のスコア
APクラス	ウィンドウ名	語彙	コマンド	
メール	シェル1	エルエス	Key("ls\n")	0
シェル	シェル2	クリア	Key("clear\n")	0
エディタ	シェル3	:	:	:

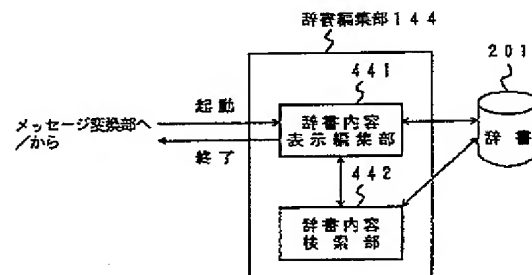
【図102】

応用プログラム ID	音声 ウィンドウ	ウィンドウ名 (属性)	入力マスク	音声入力 フラグ	認識語彙リスト	グループ ウィンドウ	親子 属性
0 (SIM)	0	システム (2)	認識結果 通知マスク	0	終了(0)	-	
	1	シェルツール(2)	"	1	エルエス(0) プロセス(0)	3	
	2	設定 (1)	"	0	オーケー(0) キャンセル(0)	-	1
	3	エディタ (2)	"	0	カット(0) コピー(0) ペースト(0)	-	
	4	設定 (1)	"	0	オーケー(0) キャンセル(0)	-	
1 (メールツール)	5	メールツール(2)	"	0	終了(0) プロセス(0) キャンセル(0) 終了(2)	-	

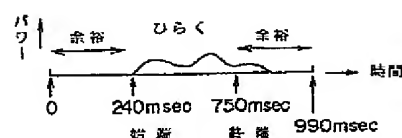
【図107】



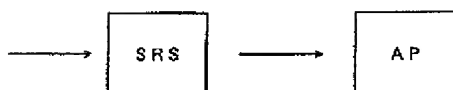
【図109】



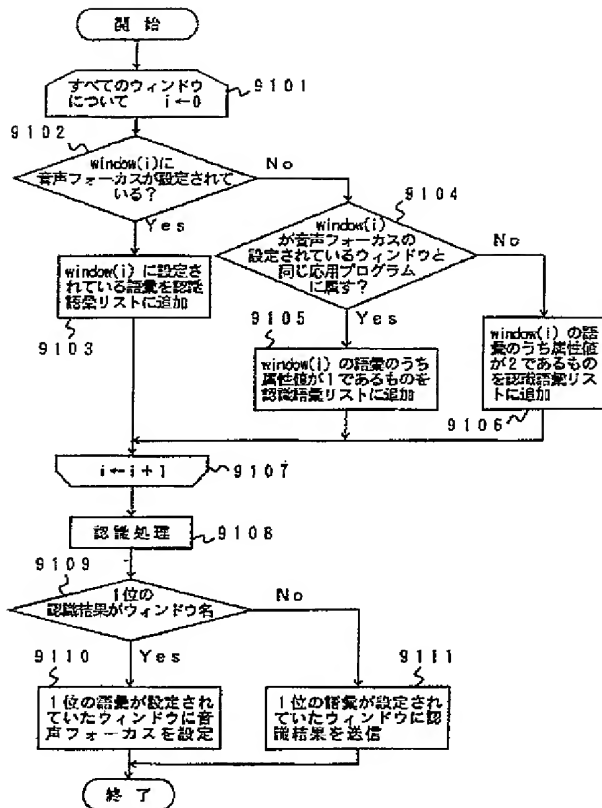
【図120】



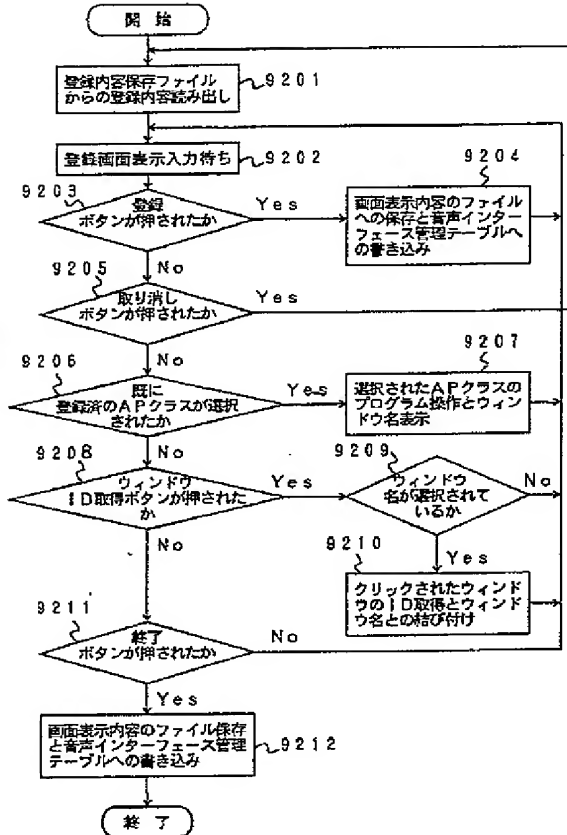
【図123】



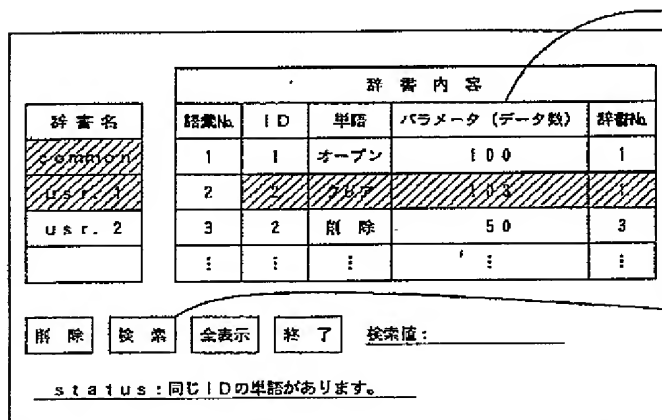
【図103】



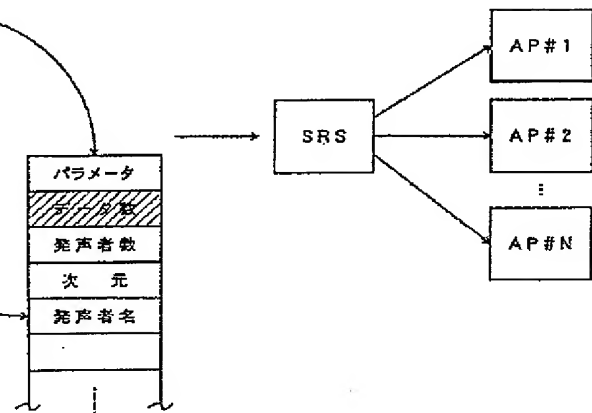
【図106】



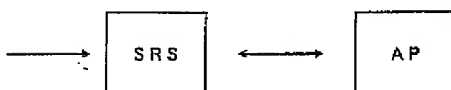
【図110】



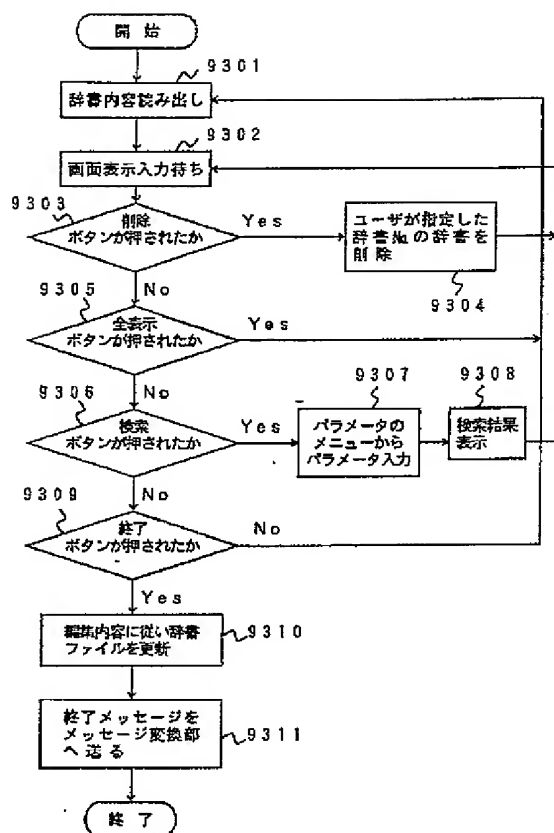
【図125】



【図124】



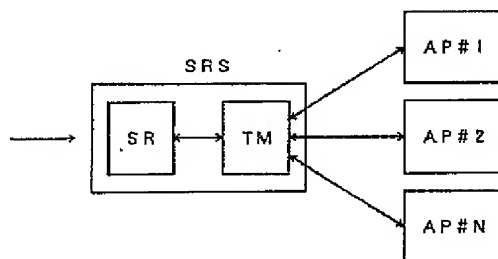
【図 1 1 1】



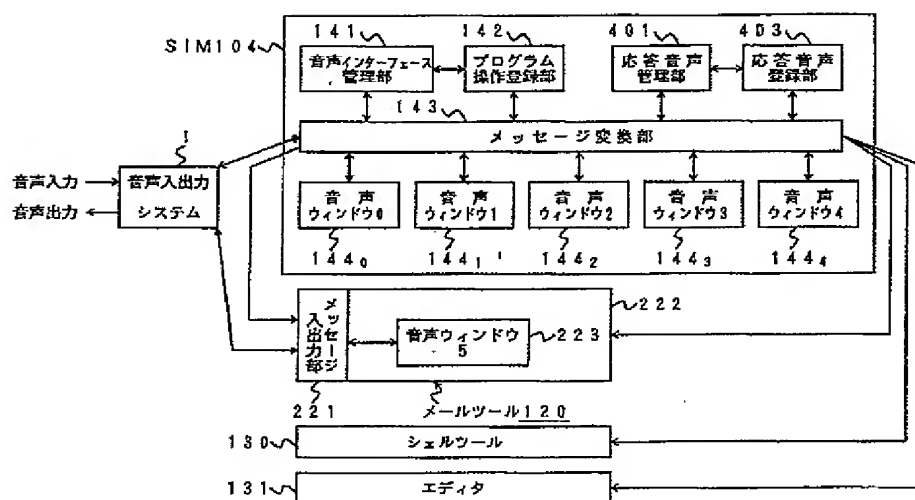
【例 1 1 3】

適用	動作	応答
×	認識成功	synth ("\$<cat>")
○	認識失敗	play ("eh. snd")
○	応用プログラム名認識	synth ("はい、\$<cat>です")
○	認識開始	synth ("認識を再開しました")
○	認識中止	synth ("認識を中止しました")
×	応用プログラム起動	play ("app-start. snd")
×	終了	play ("app-end. snd")
○	アイコン化	play ("iconify. snd")
×	アイコンを開く	play ("deiconify. snd")

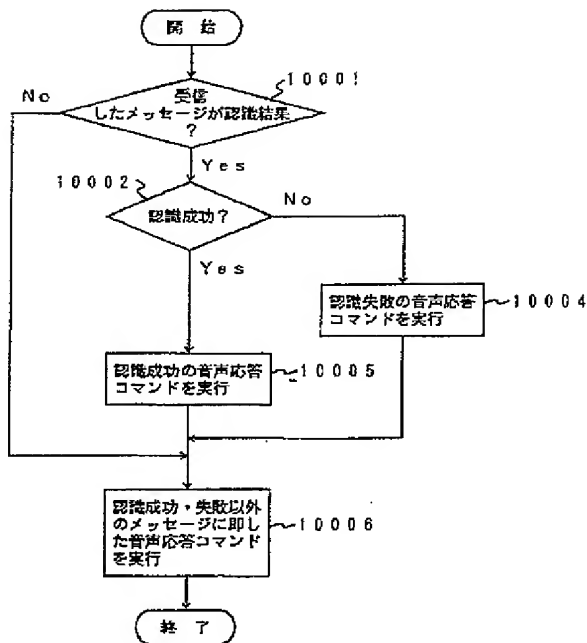
【图 126】



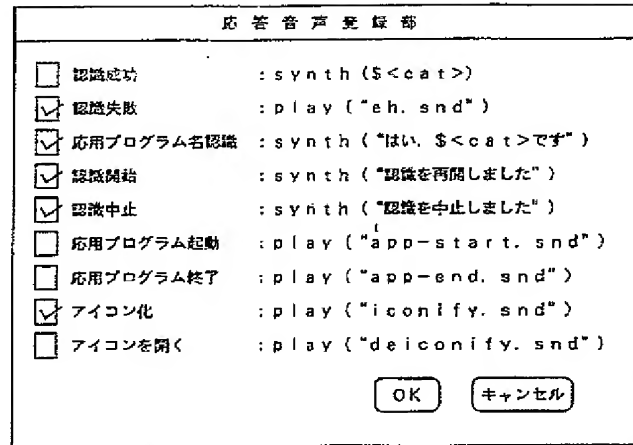
【図 1 1 2】



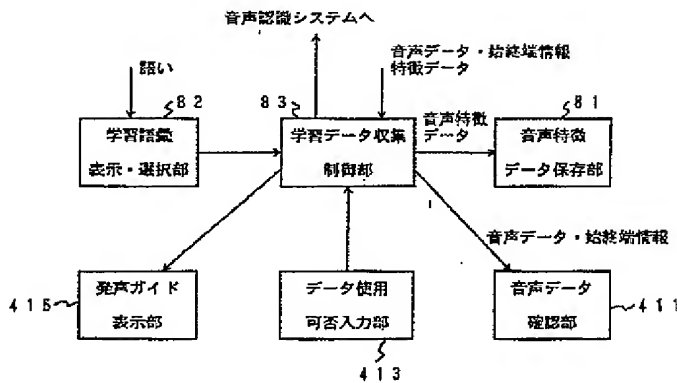
【図114】



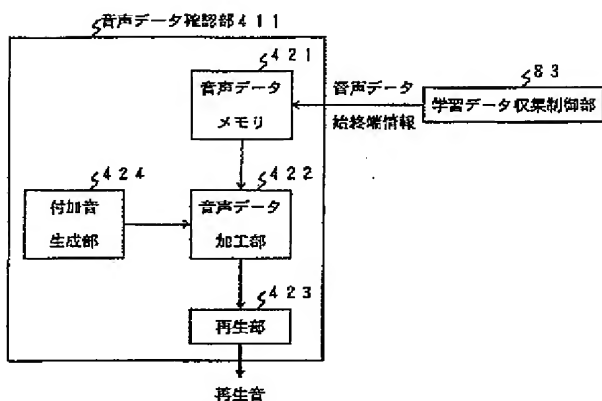
【図115】



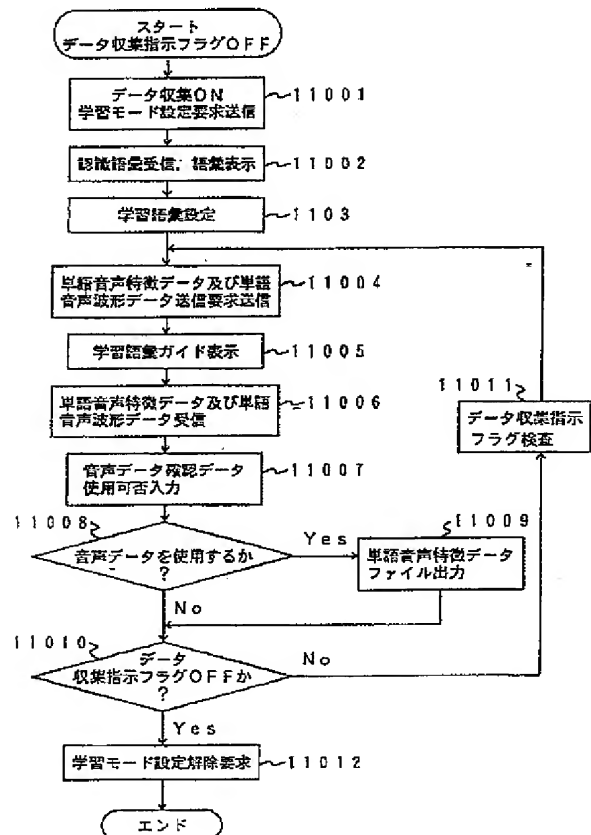
【図116】



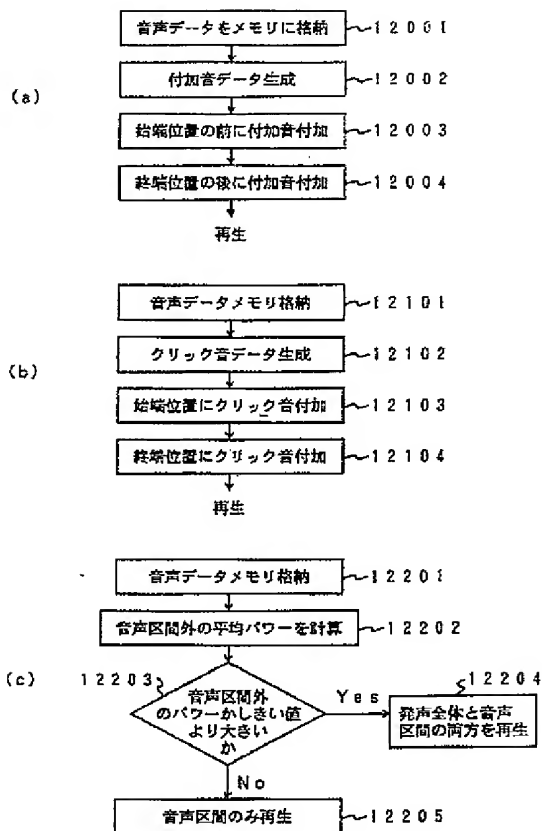
【図118】



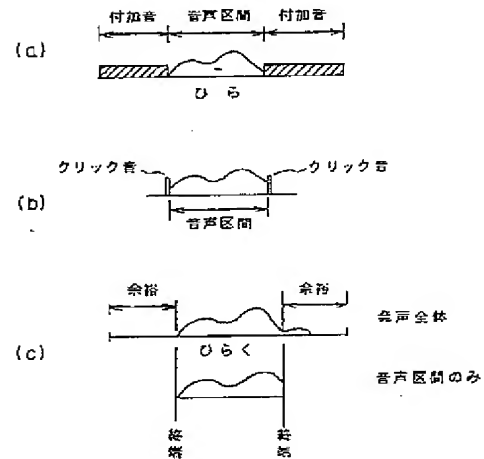
【図117】



【図119】



【図121】



フロントページの続き

(72)発明者 永田 仁史
神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内

(72)発明者 瀬戸 重宣
神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内

(72)発明者 竹林 洋一
神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内

(72)発明者 山口 浩司
神奈川県川崎市幸区小向東芝町1番地 株
式会社東芝研究開発センター内

(72)発明者 新地 秀昭
東京都青梅市新町1385番地 東芝ソフトウ
ェアエンジニアリング株式会社内